Copy 10 of 39 copies

AD-A175 217

IDA PAPER P-1966

MONTE CARLO LAYERED DEFENSE MODEL

Jerome Bracken

September 1986

This document has been approved for public release and sale; its distribution is unlimited.

INSTITUTE FOR DEFENSE ANALYSES
1801 N. Beauregard Street, Alexandria, Virginia 22311

86 12 03 079 IDA Log No. HQ 86-31571

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY DD FORM 254 DATED 1 OCTOBER 1983	3. DISTRIBUTION/AVAILABILITY OF REPORT This document is unclassified and suitable for public release.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) IDA P-1966	5. MONITORING ORGANIZATION REPORT NUMBER (S)
6a. NAME OF PERFORMING ORGANIZATION Institute for Defense Analyses; 6b. OFFICE SYMBOL (if applicable)	7a. NAME OF MONITORING ORGANIZATION N/A
6b. ADDRESS (CITY, STATE, AND ZIP CODE) 1801 North Beauregard Street Alexandria, Virginia 22311	7b. ADDRESS (CITY, STATE, AND ZIP CODE) N/A
8a. NAME OF FUNDING/SPONSORING ORGANIZATION N/A; 8b. OFFICE SYMBOL	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER IDA Independent Research
8c. ADDRESS (City, State, and Zip Code) N/A	10. SOURCE OF FUNDING NUMBERS: PROGRAM ELEMENT; PROJECT NO.; TASK NO.; ACCESSION NO. WORK UNIT

11. TITLE (Include Security Classification)
MONTE CARLO LAYERED DEFENSE MODEL

12. PERSONAL AUTHOR(S).
Jerome Bracken

13. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Final	FROM TO	1986 September	93

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Ballistic Missile Defense, Layered Defense, Strategic Defense Initiative, Boost-Phase Defense, Midcourse Defense, Terminal Defense, Preferential Defense, Monte Carlo

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

The Monte Carlo Layered Defense Model includes boost-phase defense, midcourse defense and terminal defense. It accounts for every offensive missile, RV, heavy decoy and light decoy. For near-perfect defense of a value target data base, the accounting for each RV through the succession of layers is critical, since it is not possible in advance to know if leakage or exhaustion will be the source of kill when the final events involve small integer interactions.

The model is quite flexible. In the boost-phase layer, the options for battle management/command, control and communications (BM/C^3) include random (decentralized), efficient (centralized), and preferential according to number of RVs on the missiles. In the midcourse layer, the BM/C^3 options include random, efficient, and preferential, in which the attack is shaped such that it can be handled as well as possible by the terminal defense (in the heuristic sense, not in the guaranteed optimal sense). In the terminal layer a number of different attack and defense situations are represented.

The example presented in the documentation gives test results for many combinations of defense force levels and BM/C^3 capabilities.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☐ UNCLASSIFIED/UNLIMITED ☒ SAME AS REPORT ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL	22b. TELEPHONE (Include Area Code); 22c. OFFICE SYMBOL

DD FORM 1473, 84 MAR

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

IDA PAPER P-1966

MONTE CARLO LAYERED DEFENSE MODEL

Jerome Bracken

September 1986

IDA

INSTITUTE FOR DEFENSE ANALYSES

IDA Independent Research Program

PREFACE

This study was conducted as part of the Independent Research Program of the Institute for Defense Analyses, under which significant issues of general interest to the defense research community are investigated.

TABLE OF CONTENTS

FIGURES

Table of Contents (Cont'd)

SCOPE

The Monte Carlo Layered Defense Model consists of six major parts. They are summarized below.

1. Set up target data base, including the value of each target and the number of terminal interceptors at each target.

2. Set up attack allocation. Assign the RVs of all of the attacking missiles to targets. Equip each missile with heavy and light decoys if desired.

3. Perform boost-phase defense, using one of three possible BM/C^3 weapon assignment schemes:

   a. Random Assignment: Each boost-phase defender acts independently of the others. The defender picks a missile target and shoots at it. Thus each missile may receive from none to many shots. This may be thought of as unordered fire, fully decentralized BM/C^3, or informationless fire.

   b. Efficient Assignment: Boost-phase defenders are allocated as uniformly as possible across their missile targets. For instance, if there are 1500 missiles and 1000 boost-phase interceptors, 1000 of the 1500 missiles, selected at random, receive one shot each. If there are 1500 missiles and 2000 boost-phase interceptors, 1000 missiles selected at random receive one shot each and the other 500 missiles receive two shots each. This may be thought of as ordered fire, centralized BM/C^3, or uniform simultaneous m-on-n defense.

   c. Proportional Assignment: Boost-phase defenders are allocated to missiles in proportion to the number of RVs on the missiles.

4. Perform midcourse defense, using one of three possible BM/C^3 weapon assignment schemes:

   a. Random Assignment: Analogous to boost-phase defense, discussed above.

b. Efficient Assignment: Analogous to boost-phase defense, discussed above.

c. Preferential Assignment: Midcourse defenders are assigned to RVs to shape the attack so that the RVs reaching targets chosen to be defended are likely to be killed by the terminal interceptors. To conserve the limited numbers of midcourse defenders, the algorithm proceeds as follows: First, all targets receiving less than a specified upper limit of RVs per terminal interceptor are defended so that the RVs heading for the targets are reduced below a specified lower limit. Second, the upper limit is increased and another set of targets is defended. Eventually, the lower limit is reduced if there are enough midcourse interceptors. The algorithm has not been proven to be optimal, but it is in the spirit of one known to be optimal for adaptive terminal preferential defense.

(Important Note: It turns out that in cases where the number of midcourse interceptors is relatively small and damage is moderate, preferential assignment achieves significant advantages over efficient assignment. However, when the number of midcourse interceptors is increased and damage becomes very small, preferential assignment is not superior to efficient assignment).

Decoys are included in the midcourse defense as follows. For any of the above three cases, when it is decided to attack an RV with an interceptor, the RV is examined in the context of all of the objects from its missile. The capability of the defense to discriminate RVs from decoys is represented by input probabilities of (1) missed detection against RVs, (2) false alarm against heavy decoys, and (3) false alarm against light decoys. A random process is used to either (1) attack the RV, (2) fail to attack the RV because of missing a detection, or (3) waste an interceptor on a decoy.

5. <u>Perform terminal defense,</u> using any of four possible BM/C^3 weapon assignment schemes:

a. Preallocated Fixed-Salvo Assignment: This is an implementation of the firing process associated with the Prim-Read defense. This logic is consistent with the assumption that at the individual targets the attacks are sequential and of unknown size. The defender decides in advance to fire a volley of one or more interceptors at each incoming RV. If one of the interceptors in the volley kills the RV, it does not penetrate. The number of interceptors scheduled to be fired at all of the incoming RVs adds up to the total number defending the target. For instance, if there are 10 interceptors at a target, preallocated to fire at RVs number 1 through 6, such a firing pattern might be 3,2,2,1,1,1 with no interceptors fired at RV number 7. Thus RV number 7 will surely kill the defended target, and any of the engagements against RVs 1 through 6 might also result in the defended target being killed.

b. Efficient Assignment: Analogous to boost-phase defense and mid-course defense, discussed above. Consistent with simultaneous attacks of known size at the individual targets.

c. Limited Shoot-Look-Shoot: The defender has a certain number of interceptors at the target. At the first RV he fires interceptors one-by-one until either the RV is killed or the upper bound on interceptors able to be fired is reached. For instance, there might be up to three shots at each RV. If all three fail to kill the RV there is a penetrator. This doctrine conserves interceptors, unlike the two doctrines above, for there are no interceptors "wasted" in salvos.

d. Unlimited Shoot-Look-Shoot: Same as limited shoot-look-shoot above, but no upper bound on number of shots at each RV. This doctrine produces the greatest possible effectiveness of a given number of terminal interceptors at a defended target.

6. <u>Assess damage to targets</u>. For each target, determine if there are one or more penetrating RVs.

There are two significant features of the model available to extend and elaborate on the above, as follows:

1. It is possible to suppress the boost-phase defense, either by killing a certain number of defenders or by permitting a certain number of missiles not to be engaged by the boost-phase defense.

2. It is possible to perform hybrid combinations of BM/C^3 rules within boost-phase defense and midcourse defense. Specifically, a certain portion of the missiles in boost-phase defense, or RV's in midcourse defense, can be engaged efficiently by a portion of the defense while the remainder are engaged randomly by a portion of the defense. This permits exploration of a partially centralized engagement followed by a partially decentralized engagement of the survivors.

KEY ASPECTS

Three key aspects of the qualitative behavior of the model are summarized below.

1. Random Versus Efficient Weapon Allocation in Boost-Phase and Midcourse Defense

Experience with the model has shown that there are very significant effects associated with random versus efficient assignment of boost-phase defenders. Completely decentralized assignment of boost-phase defenders leads to many missiles not being attacked, and thus substantially more RVs being presented to the midcourse defense.

Experience with the model has also shown that there are very significant effects associated with random versus efficient assignment of midcourse defenders. If midcourse defenders can be efficiently assigned, at a certain point every single RV is confronted by a one-on-one engagement, or a two-on-one engagement, and thus there tend to be few cases in which the terminal defenses are exhausted.

The amount of resources required to achieve near-zero damage is extremely sensitive to the BM/C^3 logic. Random assignment of boost-phase and midcourse defenders leads to twice as many boost-phase defenders and midcourse defenders being required to reduce damage to near-zero levels than does efficient assignment, in a test case.

To achieve efficient defense in boost-phase and midcourse requires that the defender's weapons be applied to missiles and to RVs sparingly. First consider boost-phase defense. Assume that each missile is to be engaged once. Assume that there is no shoot-look-shoot. If the attack is simultaneous, the centralized system must allocate all of the shooters to all of the targets uniformly. This must be done within a period of a minute or so. If the attack is sequential, say over five or ten minutes, the centralized system must allocate the shooters to the subset of targets presented in accordance with a preallocated rationing scheme. If one-half of the missiles have been launched, one-half of the defenders should be utilized (or something similar to this).

Partitioning the boost-phase defenders into subsets should not preclude efficient, as opposed to random, defense if the subsets are large enough. Each subset would be responsible for a subset of targets. Difficulties in efficient assignment would occur if the shooters and targets could not be related one-to-one. Since the shooters orbit out of range of targets, time-phased assurance that all missile targets receive about the same number of shots may be difficult, but should be stressed as a design parameter of the boost-phase defense.

Now consider midcourse defense, which takes place over a relatively long period of time. By monitoring each shooter and RV target, it should be possible to assure that each RV receives about the same number of shots. With respect to time considerations, efficient allocation seems easier in midcourse than in boost-phase. Simultaneous attack versus ragged attack does not seem to have significant timing effects in midcourse defense from an assignment point of view.

The principal problem in midcourse defense seems to be the effect of large numbers of decoys. If the K-factor, which is basically the number of standard deviations of the measuring process separating RVs from decoys, is large, then few RVs will escape detection and few interceptors will be wasted on decoys. If the K-factor is small, or if there are enough light decoys with even a small probability of being engaged (say 0.1), then the midcourse interceptors may be exhausted by the decoys, and the efficient allocation of defenders to RVs will fail.

2. Terminal Defense

The function of the terminal defense is to cope with the RVs which get through to attack the target. The number of RVs may vary widely, since if there are many missiles shooting at the target there may be any number from zero to a large number of RVs heading for the target after the boost-phase defense. The first two types of midcourse defense will not eliminate this phenomenon. The third type will solve it for some targets, but other targets will be heavily attacked since they are "written off" by the midcourse defense.

Performance of the terminal defense in situations where the number of defenders is just a bit larger than the number of attacking RVs provides the key difference in effectiveness. For instance, if there are seven defenders with P_k=.9, and six attackers, a preallocated defense with firing doctrine 2,2,1,1,1,0 would yield no chance of survival, while shoot-look-shoot would yield a 50 percent chance of survival. The Monte Carlo process results in quite different engagements at targets, of which a significant number are in the regions where firing doctrines make a big difference.

3. Expected Value Versus Monte Carlo

Another key aspect of the overall performance of the model is in the difference between an expected value model and a Monte Carlo model.A typical expected value model makes an assessment of the expected number of missiles killed by the boost-phase defense and then counts the expected number of RVs left. It assumes that these RVs are targeted proportional to the targeting of the missiles from whence they came. Then the expected value model makes an assessment of the number of RVs killed by the midcourse defense. The RVs surviving are again assumed to be targeted proportional to the original targeting of the missiles. These RVs are engaged by the terminal defenders. Typically, the terminal defenders significantly outnumber the RVs at all of the targets because the RVs are uniformly allocated to the targets, and thus only one or two RVs per target are confronted by at least one interceptor each. The Monte Carlo model, on the other hand, results in a significant variations in RVs appearing at the attacked targets. This one property is the main reason for the necessity for the Monte Carlo model. Some experimental results have shown that, when the P_k of boost-phase, midcourse and terminal defense is .9, the expected percent of the data base destroyed in the Monte Carlo Layered Defense Model is 5.1 percent for preallocated terminal defense and 2.1 for efficient terminal defense, whereas the expected percent of the data base destroyed by an expected value calculation is 1.0 percent and .3 percent, respectively.

For near-perfect defense, or the assured survival mission of SDI, these differences are crucial.

SUBROUTINES

1. <u>Program MAIN</u>

a. The main indexes are I1, I2, and I3, where

(1) I1 indexes on the number of boost-phase interceptors

(2) I2 indexes on the number of midcourse interceptors

(3) I3 indexes on the number of combinations of boost-phase and midcourse BM/C^3 options

b. For each value of I3, three indicators are set:

(1) I31--indicator for one of three boost-phase defense options

(2) I32--indicator for one of three midcourse defense options

(3) I33--indicator for one of four terminal defense options

The options are discussed later. In addition, the indicators IBPM and IMCM are set:

(1) IBPM--indicator for hybrid boost-phase defense, where 0 means no hybrid and 1 means hybrid

(2) IMCM--indicator for hybrid midcourse defense, where 0 means no hybrid and 1 means hybrid

c. The long print indicator LPI is set:

(1) LPI=0 means no long printout

(2) LPI=1 means long printout

d. The random number seed IRS for the entire simulation is set. The seed is in COMMON and gives a new value each time a random number is generated anywhere in the model.

e. The value target data base is set up by calling Subroutine VALUES(NVT). Subroutine VALUES creates array IV(I), I=1,NVT consisting of the values of targets 1 through NVT. It provides NVT to MAIN. Array IV is in COMMON.

f. The main loop on the number of replications, indexed by IMAIN from 1 through NUMB, is begun by the following statement:

DO 8000 IMAIN = 1, NUMB

g. The terminal defense is set up by calling Subroutine TDINV (NVT). Subroutine TDINV creates array ITD (I), I=1,NVT consisting of the number of terminal interceptors at targets 1 through NVT. Array ITD is in COMMON.

h. The attack allocation is set up by calling Subroutine ATTALL (NMIS, MNWT). Subroutine ATTALL creates arrays as follows:

IAT (I,J),	I=1,NMIS,	J=1,MNWT
IAW (I, J),	I=1,NMIS,	J=1,MNWT
IAD(I,J,1)	I=J,NMIS,	J=1,MNWT
IAD(I,J,2)	I=1,NMIS,	J=1,MNWT

NMIS is the total number of missiles and MNWT is the maximum number of targets which each missile attacks. For instance, NMIS would be 1500 and MNWT would be 2 when there are 1500 missiles, each attacking no more than two different targets.

IAT(I,J) contains the target or targets to which the RVs of missile I are assigned. For instance, if missile 10 is assigned to targets 100 and 200, then IAT(10,1)=100 and IAT(10,2)=200.

IAW(I,J) contains the number of RVs assigned from missile I to the targets specified by IAT(I,J). In the above example, if missile 10 has 8 RVs, and if four each are assigned to targets 100 and 200, then IAW (10,1) = 4 and IAW(10,2) = 4.

IAD (I, J, 1) and IAD (I, J, 2) contain the number of decoys from missile I to the targets specified by IAT (I, J). In the above example, if missile 10 has 4 heavy decoys and 50 light decoys split evenly between the two targets then IAD (10, 1, 1) = 2, IAD (10, 1, 2) = 25, IAD (10, 2, 1) = 2 and IAD (10, 2, 2) = 25.

Arrays IAT, IAW and IAD are in COMMON. The values of NMIS and MNWT are specified in Subroutine ATTALL and returned to MAIN.

i. The boost-phase defense is activated against the missiles by calling Subroutine BPD(NSHT, NTAR, MNWT, PK, IND), where:

NSHT is the number of boost-phase defenders

NTAR is the number of missile targets (typically set equal to NMIS)

MNWT is the maximum number of value targets attacked by each missile.

PK is the boost-phase defense single-shot probability of kill.

IND is the defense option, as follows:

(1) IND = 1 Random allocation of boost-phase defenders. Each boost-phase defender selects and fires at missile independently of the other defenders.

(2) IND = 2 Efficient allocation of boost-phase defenders. Boost-phase defenders are allocated uniformly over missiles.

(3) IND = 3 Proportional allocation of boost-phase defenders. Boost-phase defenders are allocated to missiles in proportion to the number of RVs on the missiles.

The indicator for the hybrid boost-phase defense IBPM is queried before Subroutine BPD is called:

(1) If IBPM = 0, then the number of shots equals NSHT, as above.

(2) If IBPM = 1, then the boost phase engagement proceeds in two steps. In the first step a certain number of shoots is assigned to efficient defense (IND = 2). In the second step the remainder is assigned to random defense (IND = 1).

j. The midcourse defense is activated against the RVs by calling Subroutine MCD (NVT, NSHT, NTAR, MNWT, PK, IND), where:

NVT is the number of value targets being protected

NSHT is the number of midcourse defenders

NTAR is the number of missiles from which the RVs are being released (typically set equal to NMIS)

MNWT is the maximum number of value targets attacked by each missile

PK is the midcourse defense single-shot probability of kill

IND is the defense option as follows:

(1) IND = 1 Random allocation of midcourse defenders. Each midcourse defender selects and fires at RVs independently of the other defenders.

(2) IND = 2 Efficient allocation of midcourse defenders. Midcourse defenders are allocated uniformly over RVs.

(3) IND = 3 Preferential allocation of midcourse defenders. Assuming knowledge of the complete attack, including the destination of all RVs, midcourse interceptors are allocated to shape the attack such that the RVs which remain after the midcourse defense can be destroyed by the terminal defense, the overall goal being to preserve as much surviving value as possible.

The indicator for the hybrid midcourse defense IMCM is queried before Subroutine MCD is called:

(1) If IMCM = 0, then the number of shooters =NSHT, as above.

(2) If IMCM = 1, then the midcourse engagement proceeds in two steps. In the first step a certain number of shooters is assigned to efficient defense (IND = 2). In the second step the remainder is assigned to random defense (IND = 1).

k. The terminal defense is activated against RVs arriving at each target by calling Subroutine TD (NVT, NMIS, MNWT, PK, IND), where

NVT is the number of value targets being protected

NMIS is the number of missiles in the attack

MNWT is the maximum number of value targets attacked by each missile

PK is the terminal defense single-shot probability of kill

IND is the defense option, as follows:

(1) IND = 1 Preallocated fixed-salvo defense. For each target, each arriving RV, sequentially, receives a salvo of a pre-specified number of interceptors.

(2) IND = 2 Efficient allocation of terminal interceptors. For each target, interceptors are allocated as uniformly as possible over all attacking RVs.

(3) IND = 3 Limited shoot-look-shoot. For each target, RVs are engaged one-by-one. The first RV is shot at by one interceptor, then another, until either it is killed or the upper limit per RV is reached. The second RV is engaged similarly. The process terminates when there are no more RVs or no more interceptors.

(4) IND = 4 Unlimited shoot-look-shoot. Same as (3) above, but no limitation on interceptor engagements per RV.

l. After the terminal defense there are NW(I), I = 1,NVT succcessful penetrators at each target. Where NW(I) = 0 the target is not destroyed. The terminal defense subroutine TD creates the relative frequency array of successful penetrators NWRF (I).

m. Value surviving after the attack is computed by Subroutine VALSURV (NVT, IVSS), which creates the value sum surviving associated with those targets where NW (I) = 0, I = 1, NVT.

n. The average and standard deviation of value damaged for the trials made thus far (up to the current value of IMAIN) is computed and printed.

o. This is the end of the main loop numbered 8000 on the number of iterations to be performed.

p. This is the end of the main loop numbered 9999 on the specification of the case to be simulated, for a particular number of boost-phase defenders and midcourse defenders, and for a defense logic for the three layers.

2. Subroutine VALUES (NVT)

Subroutine VALUES (NVT) creates the array of target values IV, where IV(I) is the value of target I, and returns to MAIN the quantity NVT, the total number of targets. The content of the subroutine is at the option of the user.

For instance, if there were 1000 targets with values 1000 through 1, the contents of the subroutine would be as follows:

```
      NVT = 1000
      DO 10 I = 1, 1000
      IT = 1001 - I
   10 IV(I) = IT
```

Subroutine VALUES (NVT) of the example, contained in Appendix B, generates a set of values for 2533 targets.

3. Subroutine TDINV (NVT)

Subroutine TDINV (NVT), Terminal Defense Inventory, creates the array of terminal inventories of interceptors ITD, where ITD (I) is the number of terminal interceptors at target I. The content of the subroutine is at the option of the user.

For instance, if in subroutine VALUES there were created 1000 targets with values 1000 through 1, and if they were to be defended by one interceptor per ten units of value, truncated to the nearest integer, the contents of the subroutine would be as follows:

```
DO 10  I = 1, 1000
10  ITD (I) = IV(I) / 10
```

Subroutine TDINV (NVT) of the example, contained in Appendix B, deploys terminal interceptors for 2533 targets equal to one-fourth of their value, truncated to the nearest integer.

4. Subroutine ATTALL(NMIS, MNWT)

Subroutine ATTALL(NMIS, MNWT), Attack Allocation, contains the allocation of attackers to targets.

It first sets the number of missiles NMIS and the maximum number of targets attacked by each missile MNWT (equivalently, maximum number of targets cross-targeted by any attacking missile).

The subroutine creates arrays IAT (I,J), IAW (I,J), IAD (I,J,K) where I = 1, NMIS, J = 1, MNWT and K = 1,2.

Consider a simple example of 1000 missiles attacking targets 1 through 1000 with 5 RVs on each missile, 2 heavy decoys on each missile and 10 light decoys on each missile. Each missile attacks only one target. The contents of the subroutine would be as follows:

```
    NMIS = 1000
    MNWT = 1
    DO 10  I = 1, 1000
    IAT (I, 1) = I
    IAW (I, 1) = 5
    IAD (I, 1, 1) = 2
10  IAD (I, 1, 2) = 10
```

Subroutine ATTALL (NMIS, MNWT) of the example, contained in Appendix B, sets up an attack on 500 targets. It allocates 500 ICBMs with 5 RVs each to the 500 targets, then 500 ICBMs with 5 RVs each to

the same 500 targets, then 500 SLBMs with 10 RVs each to the same 500 targets. In the option where decoys are included, the ICBMs have 3 RVs, 4 heavy decoys and 30 light decoys while the SLBMs have 6 RVs, 8 heavy decoys and 30 light decoys.

5. Subroutine BPD (NSHT, NTAR, MNWT, PK, IND)

Subroutine BPD (NSHT, NTAR, MNWT, PK, IND), Boost-Phase Defense, receives from MAIN the number of boost-phase shooters NSHT, number of missile targets NTAR, maximum number of targets attacked by any missile MNWT, probability of kill PK and indicator of defense option IND.

Its function is to take the array of missiles accounted for in IAT(I, J) and to kill certain of them. For those killed, the arrays in COMMON IAT (I, J), IAW (I, J), IAD (I, J, 1) and IAD (I, J, 2) are set to zero.

a. Boost-Phase Defense Suppression

The first step is to decide whether or not to suppress the boost-phase defense, and, if so, how it is to be suppressed. The indicator IBPDS is consulted, with options as follows:

IBPDS = 0 no defense suppression

IBPDS = 1 reduce number of shooters by NKLD

IBPDS = 2 reduce number of shooters by NSHT *FKLD

IBPDS = 3 do not engage number of missiles NTARS

IBPDS = 4 do not engage fraction of missiles FTARS

In the beginning of the subroutine the user sets values for IBPDS, NKLD, FKLD, NTARS, FTARS. These are used as necessary, depending on the option indicated by IBPDS.

If IBPDS = 0, 1 or 2, then IBPDST = 0. For these three cases the number of boost-phase shooters is reduced by 0, NKLD or NSHT*FKLD.

If IBPDS = 3 or 4, then IBPDST = 1. The fraction of missiles not to be engaged is set to FTARS.

b. Random Defense

Consider boost-phase defenders 1 through NSHT. For each defender, select a random number A. If IBPDST = 0, the missile corresponding to that number will be engaged. If IBPDST = 1, the missile corresponding to that number will not be engaged if A $\leq$ FTARS.

If the missile is to be engaged, draw a random number to see if it is killed. If it is killed, set IAT, IAW and IAD arrays equal 0 for that missile.

c. Efficient Defense

(1) Consider the case where boost-phase shooters are fewer than missile targets (NSHT < NTAR). Set INDTA (I), I = 1, NTAR = 0 to keep track of which targets have not been engaged. Consider boost-phase defenders 1 through NSHT. If IBPDST = 1, draw a random number A and do not engage missile if A $\leq$ FTARS. If a missile is to be engaged, draw a random number A to see which missile is to be engaged. Call this IMIS. If INDTA (IMIS) = 1 skip this missile for it has already been engaged. Instead, draw another random number to attempt to find a target. If INDTA (IMIS) = 0 then set INDTA (IMIS) = 1, draw a random number and determine if the missile is killed. If it is killed, set IAT, IAW and IAD to zero for that missile.

(2) Now consider the case where boost-phase shooters are equal to missile targets (NSHT=NTAR). Do not bother to set INDTA (I), I = 1, NTAR to 0, since all targets will be engaged unless defense is suppressed.

For boost-phase defenders 1 through NSHT, if IBPDST = 1, draw a random number A and do not engage missile if A $\leq$ FTARS. If missile is to be engaged, draw a random number A to

see if it is killed. If so, set IAT, IAW and IAD to zero for that missile.

(3) Now consider the case where boost-phase shooters are greater than number of missile targets (NSHT > NTAR). Set INDTA (I), I = 1, NTAR equal to 0 to keep track of targets which have been engaged.

A fraction P1 of the targets will receive IRP1 shots and a fraction (1 - P1) of the targets will receive IR shots, where

R = SHT/TAR

IR = R

FIR= IR

P1 = R - FIR

IRP1 = IR + 1

For example, if SHT = 1400 and TAR = 1000, then .4 of the targets will receive 2 shots and .6 of the targets will receive 1 shot.

First, consider the NTP1 targets which will receive IRP1 shots. Check to see if IBPDST=1. If so, and if A $\leq$ FTARS, a missile is not attacked. Otherwise, pick a missile IMIS at random. If INDTA(IMIS)=0 it can be attacked. Set INDTA(I)=1 and shoot IRP1 shots at the missile. If it is killed, set IAT, IAW and IAD to zero for that missile.

Second, consider the remainder of the targets. If INDTA(I)=0 and defense not suppressed, shoot IR shots at the missile. If it is killed, set IAT, IAW and IAD to zero for that missile.

d. Defense in Proportion to Number of RVs

Count the total number of RVs FIAWT. Compute the desired shots per warhead SPW.

(1) Consider the case where shooters are less than targets (NSHT < NTAR). Set INDTA(I)=0, I=1, NTAR to 0 to keep track of targets which have been engaged.

Pick a missile at random and count the number of RVs on it FIAWMT. The number of shots to be fired at that missile is SPW*FIAWMT, rounded to the next higher or lower integer (using a random number based on the number of shooters and targets). Shoot up to that many shots at the missile and if it is killed set IAT, IAW and IAD to zero.

(2) Consider the case where shooters are equal to or greater than targets (NSHT $\geq$ NTAR). There are two options within this case.

If at least one shot per missile (complete coverage) is not required ICMPLT=0) then fire shots at missiles in accordance with their number of RVS. In this case a missile may not be engaged it it has fewer RVs than the cutoff.

Alternatively, pass through all of the missiles, firing one shot at each. Then, utilizing the remainder, fire in proportion to the originally computed shots per warhead.

6. <u>Subroutine MCD (NVT,NSHT,NTAR,MNWT,PK,IND)</u>

Subroutine MCD (NVT, NSHT, NTAR, MNWT, PK, IND), Midcourse Defense, receives from MAIN the number of value targets NVT, number of midcourse shooters NSHT, number of RV targets NTAR, maximum number of value targets attacked by any missile MNWT, probability of kill PK and indicator of defense option IND.

The indicator IDECOYS is set at the beginning of MCD. If IDECOYS=0 there are no decoys. If IDECOYS=1 there are decoys. Associated with decoys are parameters as follows:

PMD = probability of missed detection

PFA1 = probability of false alarm against decoys of type 1

PFA2 = probability of false alarm against decoys of type 2.

The array IAWT (I,J) is created. It is set equal to IAW (I,J). The array IAWT will contain the actual number of RVs as the subroutine progresses. In making decisions some of the options will consult the array IAW (I,J), the RVs entering the subroutine, while others will consult the array IAWT (I,J), the current status of the simulation.

a. No Decoys, Random Defense

For shots 1 through NSHT, select an RV by drawing a random number, picking a missile and seeing if there were any RVs on that missile before it entered subroutine MCD. This is done by examining the array IAW (I, J).

If there is one or more RV associated with the missile, shoot at the RV. If it is killed, reduce the actual number in IAWT (I, J) down as far as zero.

In this option RVs can be killed even if already dead, since there is no coordination.

b. No Decoys, Efficient Defense

First, consider shooters equal to or less than targets (NSHT $\leq$ NTAR). For each shot, select a missile. If that missile actually contains an RV as indicated by IAWT (I, J), shoot at it, and if successful, reduce the actual RVs on the missile by 1. If that missile does not contain an RV, go on to consider another missile by again drawing a random number.

Next, consider shooters greater than targets (NSHT > NTAR). In this case plan to shoot at a fraction P1 of the targets IRP1 shots and at a fraction (1-P1) of the targets IR shots, where, as in the boost-phase defense,

$$R = SHT/TAR$$

$$IR = R$$

FIR = IR

P1 = R-FIR

IRP1 = IR+1

For each missile, determine whether or not it has any RVs. If it does, count the total IAWTT. For each of the IAWTT, by drawing a random number, decide if it will receive IRP1 or IR shots. Shoot up to the desired number of shots at the RV. If it is killed, reduce IAWT (I, J) by 1.

c. <u>No Decoys, Preferential Defense</u>

For this case Subroutine MCD3 (NVT, NSHT, NTAR, MNWT, PK) is called.

The number of RVs assigned to each target is computed as NWT (IMAIN), IMAIN=1, NVT. For each target indexed by IMAIN, the indicator INDI (IMAIN, JT)=I indicates the number of the Ith missile shooting at it and the indicator INDJ (IMAIN, JT) = J indicates the Jth group of RVs from the Ith missile. In the example there are three missiles shooting at each target, and 2533 targets, so the dimensions are INDI (2533,3) and INDJ (2533,3).

Next, there are seven iterations wherein the midcourse defense attempts to shape the entire attack so that the terminal defenders can cope with it. Upper bounds RATSU and lower bounds RATSL are set for the quantity

RAT = NWT(IMAIN)/ITD(IMAIN)

or ratio of attackers to terminal defenders. The least-attacked targets are defended first (as measured by an upper bound), reducing the RVs until they are below a certain lower bound. Then another set of controls is established, and so on. The algorithm is summarized as follows:

Count number of RVs assigned to each target from all surviving missiles. At each target let X=RVs/Terminal Defenders. Until midcourse defense is exhausted:

(1) For all targets where $X \leq 1$ kill sufficient RVs to achieve $X \leq .8$

(2) For all targets where $X \leq 2$ kill sufficient RVs to achieve $X \leq .8$

(3) For all targets where $X \leq 3$ kill sufficient RVs to achieve $X \leq .8$

(4) For all targets where $X \leq 4$ kill sufficient RVs to achieve $X \leq .8$

(5) For all targets where $X \leq U^*$ kill sufficient RVs to achieve $X \leq .8$

(6) For all targets where $X \leq .8$ kill sufficient RVs to achieve $X \leq .6$

(7) For all targets where $X \leq .6$ kill sufficient RVs to achieve $X = 0$,

where U* is the input upper bound (currently 10).

As RVs are killed they are subtracted from IAWT.

d. Decoys, Random Defense

For shots 1 through NSHT, an RV is selected by drawing a random number, picking a missile and seeing if there were any RVs on that missile before it entered subroutine MCD. This is done by examining IAW(I, J). If there is an RV in IAW (IMIS, J) the subroutine MCDWD (Midcourse Defense with Decoys) is called, with calling sequence MCDWD (IND,IAWT,IMIS, J,

MNWT, PMD, PFA1, PFA2, PK, INDSHT). The subroutine decreases IAWT (IMIS, J) if an RV is killed and sets

INDSHT=0 when no midcourse defender expended

INDSHT=1 when midcourse defender expended (either against real target or false target).

The first indicator in the calling sequence, IND, is set to 0 for random defense and 1 for efficient defense.

The subroutine MCDWD operates as follows. The total number of objects NOBJT on a missile is counted, made up of the components:

NOBJ1 = number of RVs

NOBJ2 = number of heavy decoys

NOBJ2 = number of light decoys

where

NOBJT = NOBJ1 + NOBJ2 + NOBJ3.

Each RV is treated as though it comes from a cluster characterized by the initial mix of objects as above, except when:

IND=1, NOBJ1 is computed from IAWT.

IND=0, NOBJ1 is computed from IAW

The group of objects totaling NOBJT is processed as follows. A random number is drawn and an object chosen which matches it. (For instance, if there are 4 RVs, 2 heavy decoys and 4 light decoys and the random number = .05 the object is an RV, while if the random number = .91 the object is a light decoy.

Another random number A is chosen. If the object is an RV and if A ≤ PMD (probability of missed detection) it is not shot

at. A defender is not wasted but neither is an RV engaged. If A > PMD the RV is engaged by drawing a random number A and if A ≤ PK the RV is killed.

If the object is a decoy PFA is set equal to PFA1 or PFA2, depending on the type. A random number is chosen. If A ≤ PFA there is a false alarm and midcourse defender is wasted.

e. Decoys, Efficient Defense

This portion of the program is analogous to the no decoys, efficient defense portion discussed in b. above. However, the subroutine MCDWD is called.

Before entering MCDWD, IND=1, so the number of RVs is computed from IAWT rather than IAW as in the above discussion. Otherwise, the procedure is the same as discussed in d. above.

f. Decoys, Preferential Defense

Not currently coded.

7. Subroutine TD (NVT, NMIS, MNWT, PK, IND)

Subroutine TD (NVT, NMIS, MNWT, PK IND), Terminal Defense, receives from MAIN the number of value targets NVT, number of missiles in the attack NMIS, maximum number of value targets attacked by any missile MNWT, probability of kill PK, and indicator of defense option IND.

The number of warheads arriving at each target is calculated for all targets, namely NW (IMAIN), IMAIN = 1, NVT. Then for all targets one of four attrition methods is utilized in determining the number of warheads surviving after the terminal defense, or the final status of NW (IMAIN), IMAIN = 1, NVT.

a. Preallocated Fixed-Salvo Terminal Defense

A particular doctrine is chosen for the targets defended by each number of interceptors. In the example, there are options

for 2, 5 and 10 interceptors at each target. Their firing doctrines are, respectively,

2 interceptors -- 1, 1

5 interceptors -- 2, 1, 1, 1

10 interceptors -- 2, 2, 2, 1, 1, 1, 1

For instance, with five interceptors, 2 are fired at the first RV, 1 at the second, 1 at the third, 1 at the fourth, and none thereafter. This defense should be conceptualized as fitting a situation where the attack is sequential and of size unknown to the defender. For 10 interceptors, the "price" of the target is 8, which surely kills it. The defense is balanced so that the attacker has about the same expected value per RV up to an attack size of 8 RVs. This preallocated fixed-salvo defense is analogous to the "Prim-Read" defense.

For each target IMAIN = I, NVT, the number of warheads NWT destined for it is determined. The number of engagements JENGS is chosen corresponding to the firing doctrine associated with the number of interceptors defending the target. For RV 1 through NWT the number of interceptors fired is chosen from the firing doctrine. As RVs are killed they are subtracted from NW (IMAIN).

b. Efficient Terminal Defense

At each target indexed by IMAIN = 1, NVT, there are NW (IMAIN) RVs arriving. There are ITD (IMAIN) terminal interceptors. Set NWT = NW (IMAIN) and ITDT = ITD (IMAIN).

If ITDT ≤ NWT, there are equal to or fewer interceptors than RVs. A one-on-one engagement is simulated against ITDT of the RVs. The number of successful RVs is noted.

If ITDT > NWT there are more interceptors than RVs. As in the efficient boost-phase defense and efficient midcourse defense previously discussed, there are IRP1 defenders shot at

a fraction P1 of the RVs and IR defenders shot at a fraction (1-P1) of the RVs.

Each RV killed causes NW (IMAIN) to be decremented by one. The final number of successful RVs is the remainder in NW (IMAIN) after the terminal defense.

c. Limited Shoot-Look-Shoot Terminal Defense

In this case the defender need not fire salvos at RVs. He can shoot an interceptor, observe if the RV is killed, shoot again if necessary, and so on, up to a certain limit per incoming RV. Thus he can substantially conserve defenders, and, if the upper limit on number of shots is reasonably high, have a high probability of kill against each RV.

The program is set to have the upper limit of the truncation limit of the minimum of Value/5 and 2. In the example, the value of the attacked target is either 20 or 10, so there are effectively 4 or 2 shots as an upper limit.

The number of terminal defenders is ITDT. The number of warheads heading for the target is NWT. The maximum number of shots is MAXDEF. For RVs 1 through NWT, interceptors are fired one by one up to a maximum per target of MAXDEF. If an interceptor is successful, as determined by the random number drawn being equal to or less than PK, then NW(IMAIN) is decremented by one. As each interceptor is fired ITDT is decremented by one. The process terminates when no more RVs are coming or when ITDT is 0. The final number of successful RVs is the remainder in NW(IMAIN).

d. Unlimited Shoot-Look-Shoot Terminal Defense

The logic for this defense is exactly the same as c. above, except there is no upper limit on the number of shots taken at any particular RV. There is no check of MAXDEF as in the above discussion.

e. Relative Frequency Array of Successful Penetrators

At the end of Subroutine TD a relative frequency array of successful penetrators is created. The array is denoted NWRF.

In the example there may be any number from 0 through 20 successful penetrators, so the relative frequency array NWRF has 21 elements.

8. Subroutine VALSURV (NVT, IVSS)

Subroutine VALSURV (NVT, IVSS) receives from MAIN the number of value target NVT and returns the value sum surviving IVSS. It simply sums the values IV(I), I = 1, NVT associated with the targets receiving zero successful penetrators (those with NW (I) =0, I=1, NVT).

EXAMPLE

This example is presented in order to give the reader information on the level of detail at which the model is used and of the sensitivities of the outcome to variations in BM/C^3 options and defensive resources. Most of the features of the program are exercised and illustrated.

1. Set up Targets of Differing Value

The target data base is composed of the array IV of targets of differing value. As an example, consider the following list of targets. There are eleven conceptual types of targets, with value per target as shown and total value as shown.

Type of Target	Number of Targets	Value Per Target	Total Value
SLBM Base	3	200	600
C^3 Installation	10	100	1000
ABM Base	10	50	500
Bomber Base	50	20	1000
Air Defense Base	20	20	400
CONUS Military Inst.	200	20	4000
Theater Military Inst.	200	20	4000
C^3 Installation	40	10	400
CONUS Econ. Inst.	500	10	5000
Theater Econ. Inst.	500	10	5000
ICBM	1000	5	5000
	2,533		26,900

The target data base illustrated consists of 2533 targets with a total value of 26,900.

The subroutine VALUES creates the target data base.

2. Set Up Terminal Interceptors

For each target, a certain number of terminal interceptors is deployed in array ITD. As an example, for each target in the data base given above,

interceptors per target are deployed equal to half of the value per target (rounded down to the next lower integer). The result is as follows:

Type of Target	Number of Targets	Value Per Target	Interceptors Per Target	Total Interceptors
SLBM Base	3	200	100	300
C^3 Installation	10	100	50	500
ABM Bases	10	50	25	250
Bomber Bases	50	20	10	500
Air Defense Bases	20	20	10	200
CONUS Military Inst.	200	20	10	200
Theater Military Inst.	200	20	10	2000
C^3Installations	40	10	5	200
CONUS Econ. Inst.	500	10	5	2500
Theater Econ. Inst.	500	10	5	2500
ICBMs	1000	5	2	2000
	2,533			12,950

There are 12,950 terminal interceptors assigned to the 2533 targets. The subroutine TDINV creates the terminal interceptors assignments in array ITD.

3. Set Up Attack Allocation

The example attack allocation consists of attacking the first 500 targets defended by fewer than 25 interceptors, namely targets 24 through 523. Each target is attacked by the RVs from three missiles. Target 24, for instance, is attacked by missile number 1 with 5 RVs, by missile 501 with 5 RVs, and by missile number 1001 with 10 RVs. The attack is as follows:

Missile Number	Target Number	RVs Per Target
1	24	5
.	.	.
.	.	.
.	.	.
500	523	5
501	24	5
.	.	.

.	.	.
1000	523	5
1001	24	10
.	.	.
.	.	.
1500	523	10

The above example includes cross-targeting in the assignment of more than one missile to each target. The computer program, however, is more flexible in the sense that the logic permits each RV to be cross-targeted. For instance, missile 1 could logically fire 1 RV each at targets 24, 25, 26, 27 and 28. (The program dimensions are currently limited to the targeting at most two targets by each missile, but the logic is completely flexible).

From the point of view of the targets, the above allocation results in the following assignment of RVs:

Target Number	Value Per Target	Terminal Interceptors Per Target	Attacking RVs Per Target
24	20	10	20
.	.	.	.
.	.	.	.
493	20	10	20
494	10	5	20
.	.	.	.
.	.	.	.
523	10	5	20

The salient point is that each target receives 20 RVs, from three missiles. Since the boost-phase defense kills missiles before they release RVs each target can be attacked by 0, 5, 10, 15 or 20 RVs, depending on which missiles (if any) are destroyed.

Another interpretation worth keeping in mind is that the example attack allocation might be conceptualized as assigning 1000 ICBMs with 5 RVs each, plus 500 SLBMs with 10 RVs each, to 500 relatively high value targets.

4. Results for 12,950 Terminal Interceptors

Figure 1 presents, on the left side, results for 1000 boost-phase interceptors and, on the right side, results for 2000 boost-phase interceptors. Units of value destroyed are presented as a function of number of midcourse interceptors for a wide variety of BM/C^3 options.

On the left side, graph A,A,A is for random allocation of boost-phase, random allocation of midcourse, and preallocated fixed-salvo defense at each target. It yields the most damage for the attack of 10,000 RVs. Graph A,A,D, with unlimited shoot-look-shoot, does quite a bit better, and damage is relatively small with 5000 midcourse interceptors.

Graphs B,B,A and B,B,D display much more defense effectiveness than A,A,A and A,A,D. In particular, B,B,D with 3000 interceptors yields relatively small damage. Note that B,B,D as compared with B,B,A, and A,A,D as compared with A,A,A are significantly better. The inventory of 12,950 terminal defenders with unlimited shoot-look-shoot is much more effective than when these same terminal defenders are limited to a preallocated fixed-salvo firing doctrine.

Finally, on the left side, consider B,C,D. It is the best defense with 1000 interceptors but does not do much better than B,B,D with 3000 interceptors. (There will be more discussion of preferential midcourse defense in the following section).

Turning to the right side of Figure 1, note that 2000 efficient boost-phase interceptors (B,B,A and B,B,D) achieve near-zero damage with as few as 1000 efficient midcourse interceptors. Confronting 1500 missiles with 2000 boost phase interceptor shots efficiently allocated, and confronting the remaining RVs with 1000 midcourse interceptors efficiently allocated, results in expected damage of zero for B,B,D and a very small amount for B,B,A.

Staying with the right side of Figure 1, consider the random defenses A,A,A and A,A,D. The former requires 5000 midcourse interceptors to achieve near-zero damage, while the latter achieves this with 3000 midcourse interceptors.

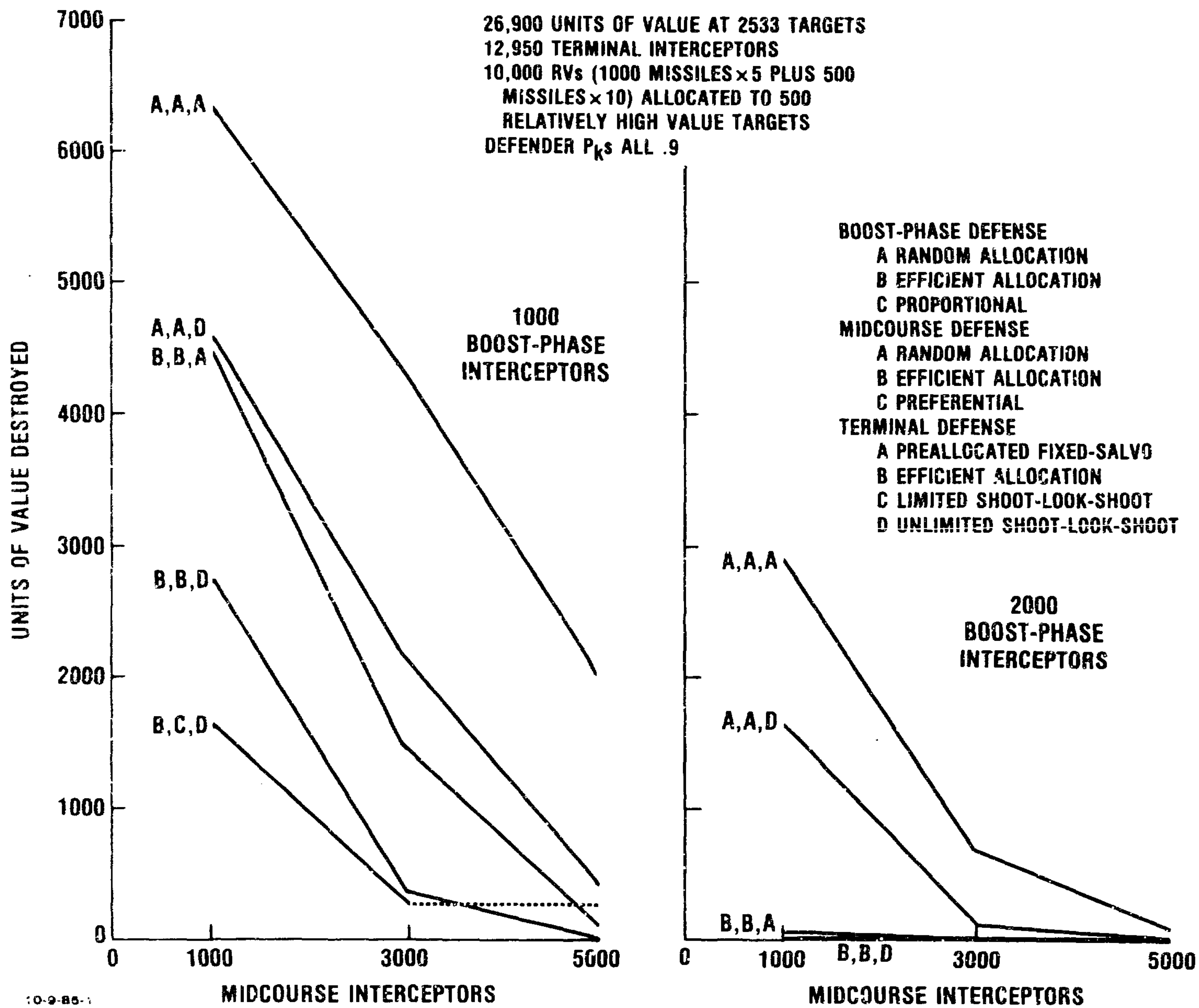

Figure 1. RESULTS FOR 12,950 TERMINAL INTERCEPTORS

Comparing the left with the right sides of Figure 1, note that 1000 boost-phase interceptors and 3000 midcourse interceptors with B,B,D does almost as well as 2000 boost-phase interceptors and 5000 midcourse interceptors with A,A,A. The latter needs twice the number of boost-phase interceptors and almost twice the number of midcourse interceptors to achieve the same effectiveness. This illustrates the importance of BM/C^3 capabilities for weapon allocation.

5. Results for 12,950 and 5,950 Terminal Interceptors

The next case is to let the number of terminal interceptors per target be equal to one-fourth of the value per target (rounded down to the next lower integer). The result is as follows:

Type of Target	Number Per Target	Value Per Target	Interceptors Per Target	Total Interceptors
SLBM Base	3	200	50	150
C^3 Installation	10	100	25	250
ABM Base	10	50	12	120
Bomber Base	50	20	5	250
Air Defense Base	20	20	5	100
CONUS Military Inst.	200	20	5	1000
Theater Military Inst.	200	20	5	1000
C^3 Installation	40	10	2	80
CONUS Econ. Inst.	500	10	2	1000
Theater Econ. Inst.	500	10	2	1000
ICBMs	1000	5	1	1000
	2,533			5,950

There are thus 5,950 as opposed to 12,950 terminal interceptors.

Figure 2 presents the results. First, on the left side, consider the solid lines for A,A,A and A,A,D. The two cases are quite close together, whereas in Figure 1

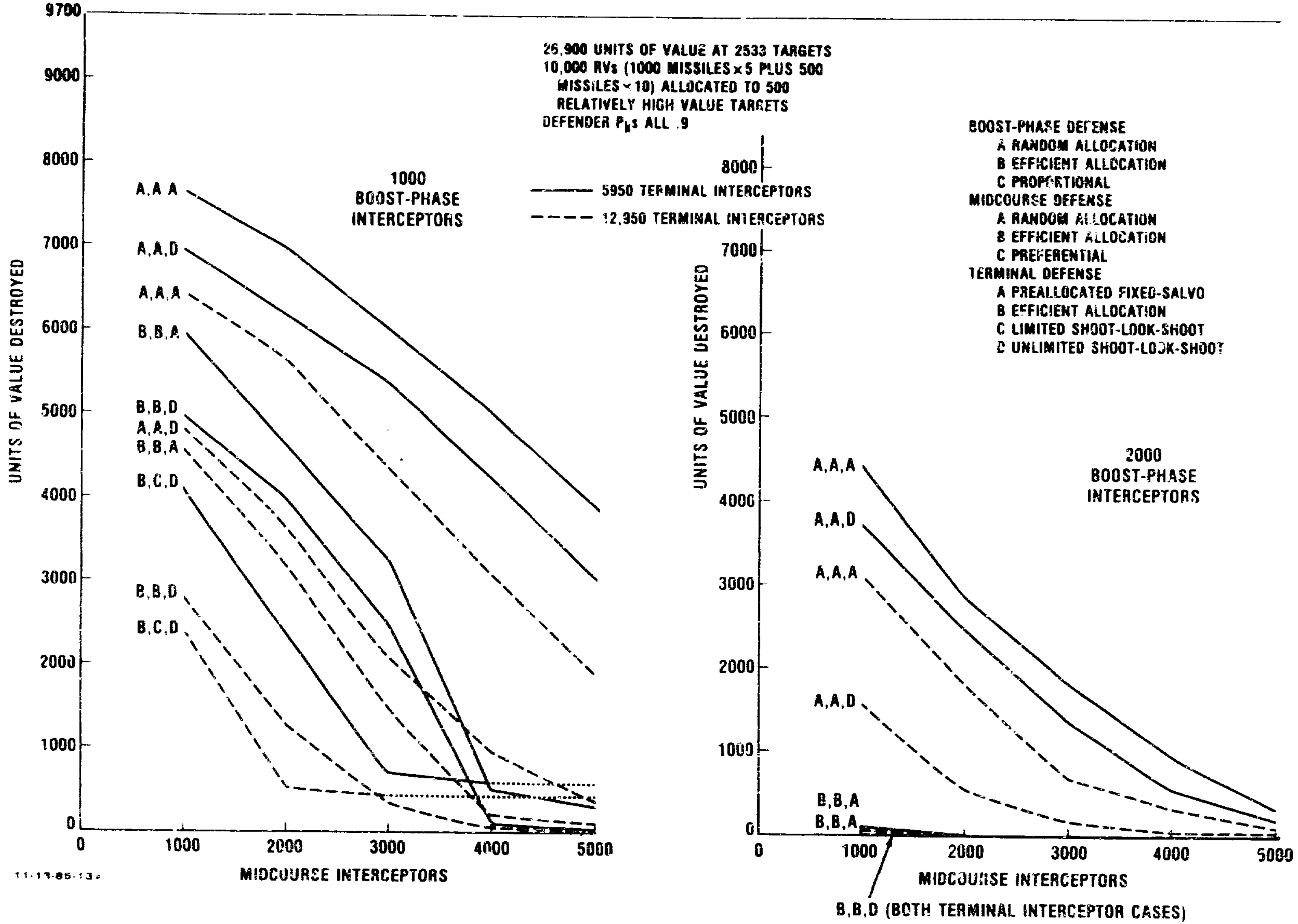

Figure 2. RESULTS FOR 12,950 AND 5,950 TERMINAL INTERCEPTORS

they had been far apart (as they are in the dashed lines of Figure 2). When there are fewer terminal interceptors, whether they are preallocated or shoot-look-shoot does not yield so much difference in payoff.

Second, on the left side, consider the solid lines for B,B,A and B,B,D. There is a dramatic impact between 3000 and 4000 midcourse interceptors. The payoff from taking on all of the RVs surviving the boost-phase defense is immense. When there are 1000 boost-phase interceptors with P_k of 0.9, there are roughly 600 surviving missiles of which 400 have 5 RVs and 200 have 10 RVs, or 4000 RVs. If these can all be engaged the number of survivors is roughly 400, distributed (quite non-uniformly) across the 500 attacked targets. Both types of terminal defenses are effective, with the unlimited shoot-look-shoot doing better. The main point, however, is that having enough efficient midcourse defenders to engage all of the RVs makes a tremendous difference.

Third, on the left side, consider the solid line for B,C,D. Up to 3000 midcourse defenders preferential midcourse defense does quite a bit better than efficient midcourse defense (B,B,D). Beyond this the algorithm does not beat the uniform defense. If the defender were to observe the attack size, he could simply switch to uniform defense if he knew the attack size exceeded 4000.

In general, the preferential defense B,C,D is a difficult optimization problem. The algorithm described previously is independent of attack size. A better algorithm would solve the large-scale integer assignment of shooters to targets knowing both numbers of shooters and targets, and knowing the terminal defense capabilities. However, the present example makes the point that preferential midcourse defense is much better than efficient midcourse defense when the number of midcourse defenders is limited.

Fourth, consider the right side of Figure 2. As before, if there are 2000 efficient boost-phase interceptors and 1000 or more efficient midcourse interceptors (B,B,A or B,B,D) there is very little damage for the solid or dashed lines. Halving the numbers of terminal interceptors is not a problem. However, if both the boost-phase and midcourse interceptors are random (A,A,A and A,A,D) the solid lines are significantly worse than the dashed ones.

In particular, A,A,D is much worse, for unlimited shoot-look-shoot with fewer terminal interceptors is negated by exhaustion at quite a few targets. Where 3000 midcourse interceptors result in near-zero damage with the dashed line, A,A,D, 5000 midcourse interceptors are required with the dashed line A,A,D.

6. Efficient Boost-Phase, Random Midcourse

The right side of Figure 3 presents results for efficient boost-phase defense followed by random midcourse defense. The dotted lines are the new results.

There are few enough missiles surviving the efficient boost-phase defense that random midcourse defense does not result in significantly more damage than efficient midcourse defense.

7. Random Boost-Phase, Efficient Midcourse

The right side of Figure 4 presents results for random boost-phase followed by efficient midcourse defense. The dotted lines are the new results.

Note that 2000 random boost-phase defenders let through so many RVs that 1000 efficient midcourse interceptors do not do much good.

However, as the numbers of efficient midcourse interceptors increases from 2000 to 3000 the attack is sufficiently diluted so that just a few targets are destroyed. This midcourse defense situation is exactly analogous to the left side of the figure increasing from 3000 to 4000 efficient midcourse interceptors.

8. Random Boost-Phase, Hybrid Midcourse

The right side of Figure 5 presents results for random boost-phase defense followed by a combination of efficient midcourse defense and random midcourse defense. The dotted lines are the results.

Hybrid defense is defined as one-half efficient defense followed by one-half random defense. (It is denoted by B' = 1/2B, 1/2A). Thus, the same number of RVs after boost-phase defense as in Figure 4 is confronted by a partially efficient and partially random midcourse defense.

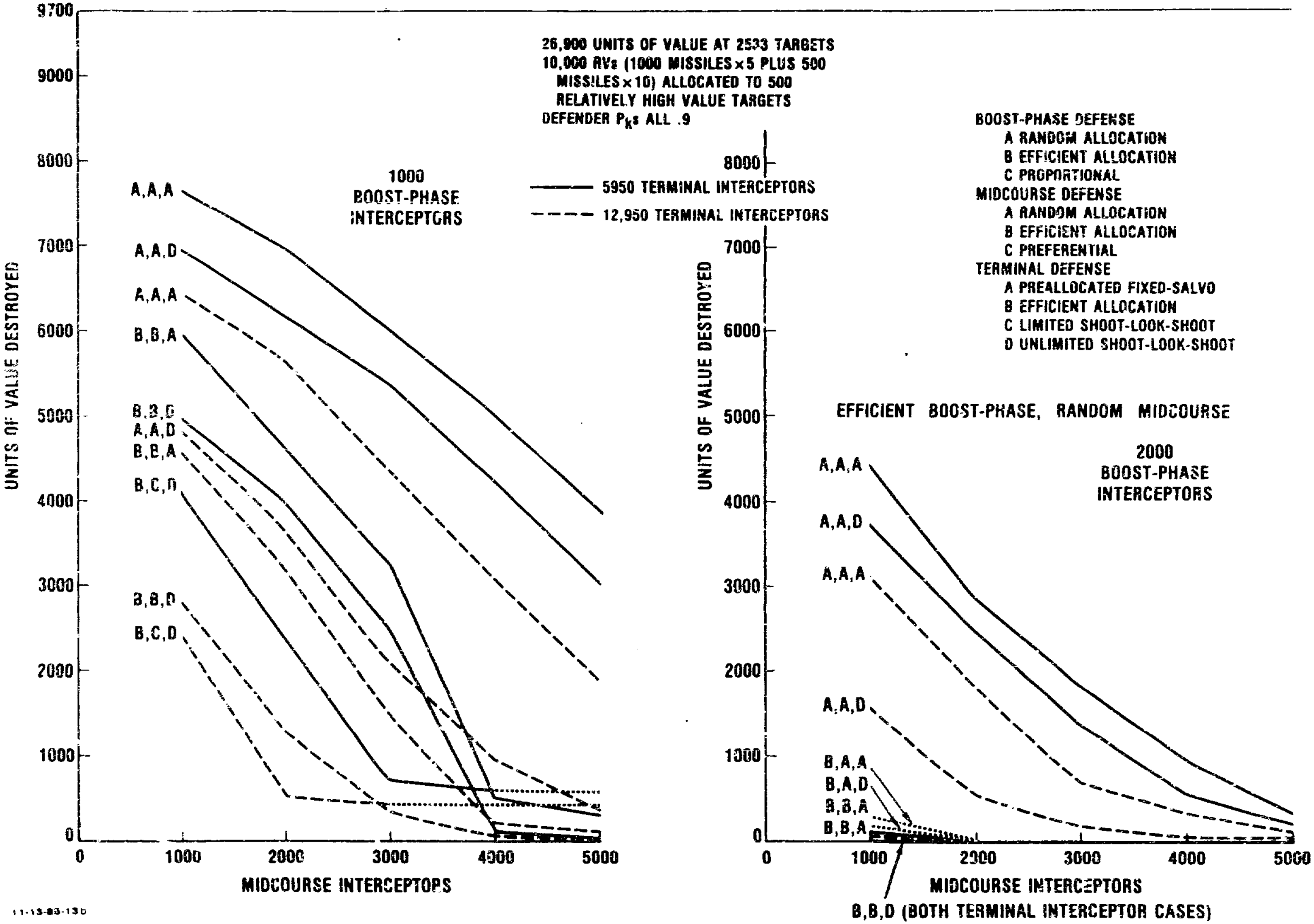

Figure 3. EFFICIENT BOOST-PHASE, RANDOM MIDCOURSE

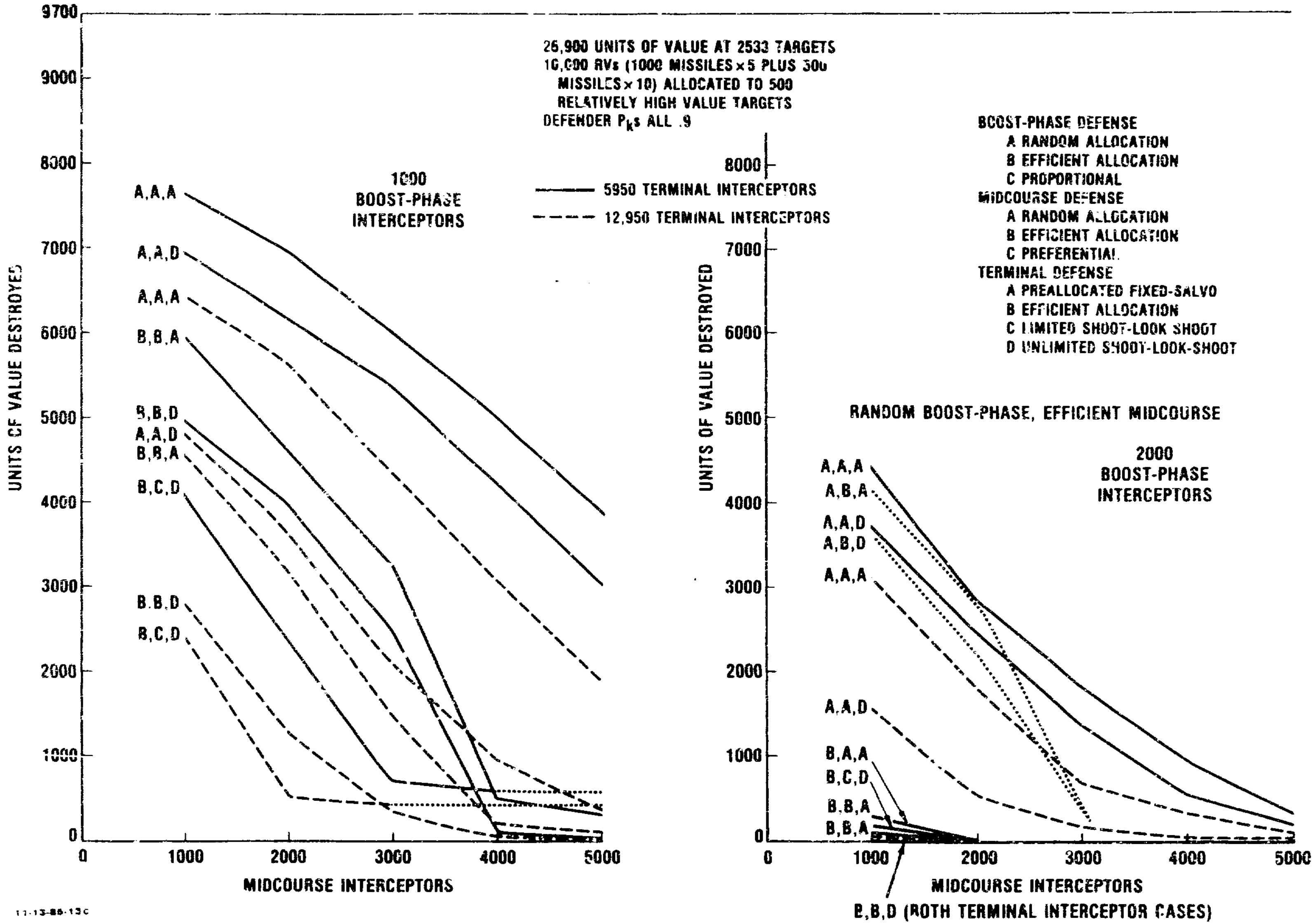

Figure 4. RANDOM BOOST-PHASE, EFFICIENT MIDCOURSE

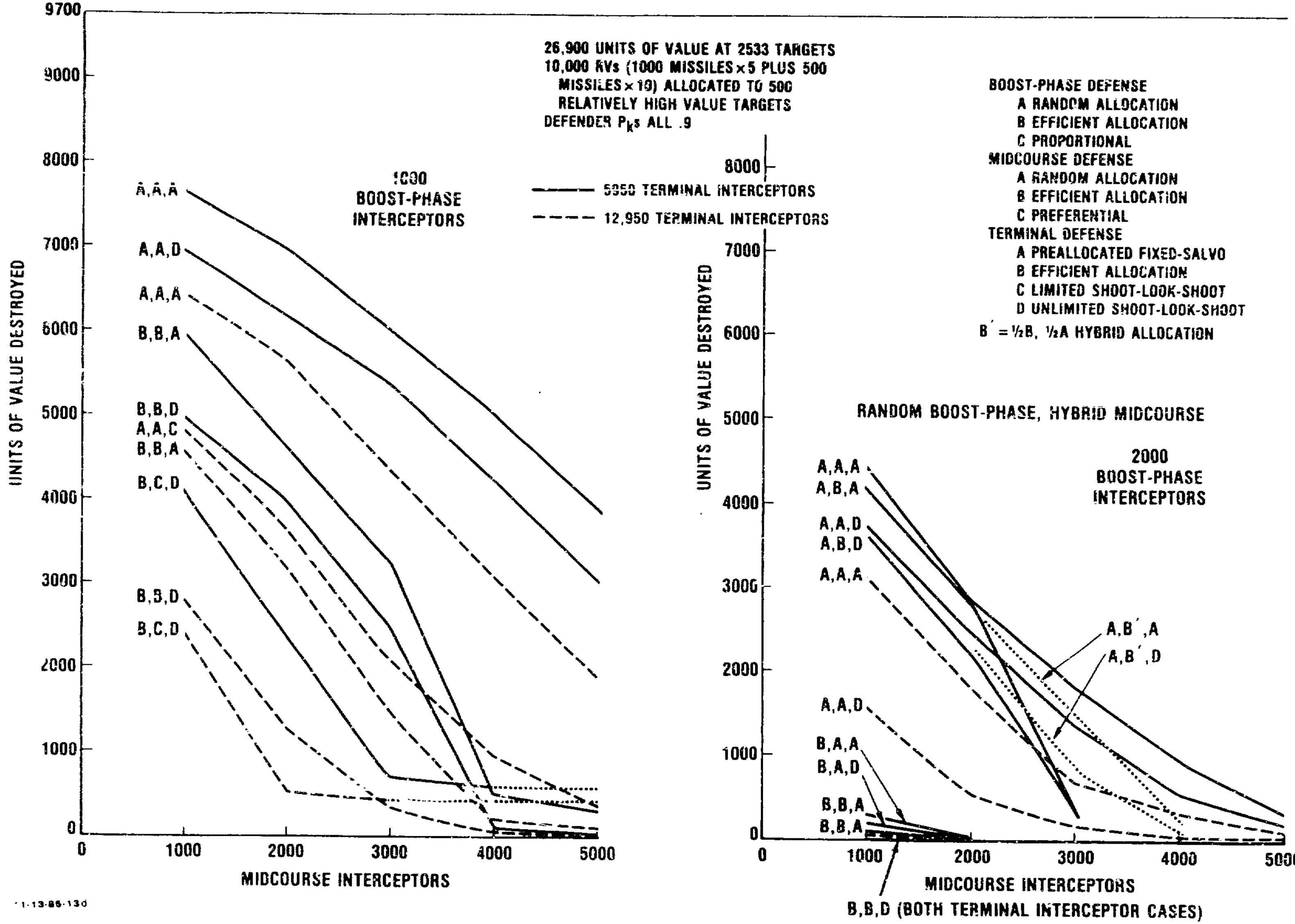

Figure 5. RANDOM BOOST-PHASE, HYBRID MIDCOURSE

The general behavior is for the units of value destroyed to decrease more gradually as a function of number of midcourse defenders. The number of midcourse defenders to yield low damage increases from 3000 with efficient defense (A,B,A and A,B,D) to 4000 with hybrid defense (A,B',A and A,B',D).

9. Hybrid Boost-Phase, Hybrid Midcourse

The right side of Figure 6 presents results for hybrid boost-phase defense followed by hybrid midcourse defense.

In this case, a hybrid boost-phase defense is composed of 1000 efficient boost-phase defenders and 1000 random boost-phase defenders. The 1000 efficient defenders engage a subset of the 1500 missiles and the 1000 random defenders engage the remaining missiles (roughly 600 missiles). Thus the number of RVs proceeding to be confronted by the midcourse defense is more than in the efficient boost-phase case but less than in the random boost-phase case.

As expected, the effects of adding hybrid midcourse interceptors are somewhat gradual, with value destroyed being fairly low with 3000 defenders. The gradual effects of the hybrid midcourse defense can be contrasted with the sharp effects of the efficient midcourse defense (see, for example, B',B',A and B',B',D versus A,B,A and A,B,D).

10. Hybrid Boost-Phase, Random Midcourse

The right side of Figure 7 presents the last excursion of this type. Hybrid boost-phase defense is followed by random midcourse defense. The effects of adding random midcourse defenders are more gradual than adding hybrid midcourse defenders.

11. Suppression of Boost-Phase Defense

The right side of Figure 8 presents results for two types of suppression of boost-phase defenses.

In the first type of suppression boost-phase defenders are reduced by one-third, or from 2000 to 1333. Not all of the missiles are engaged. When the

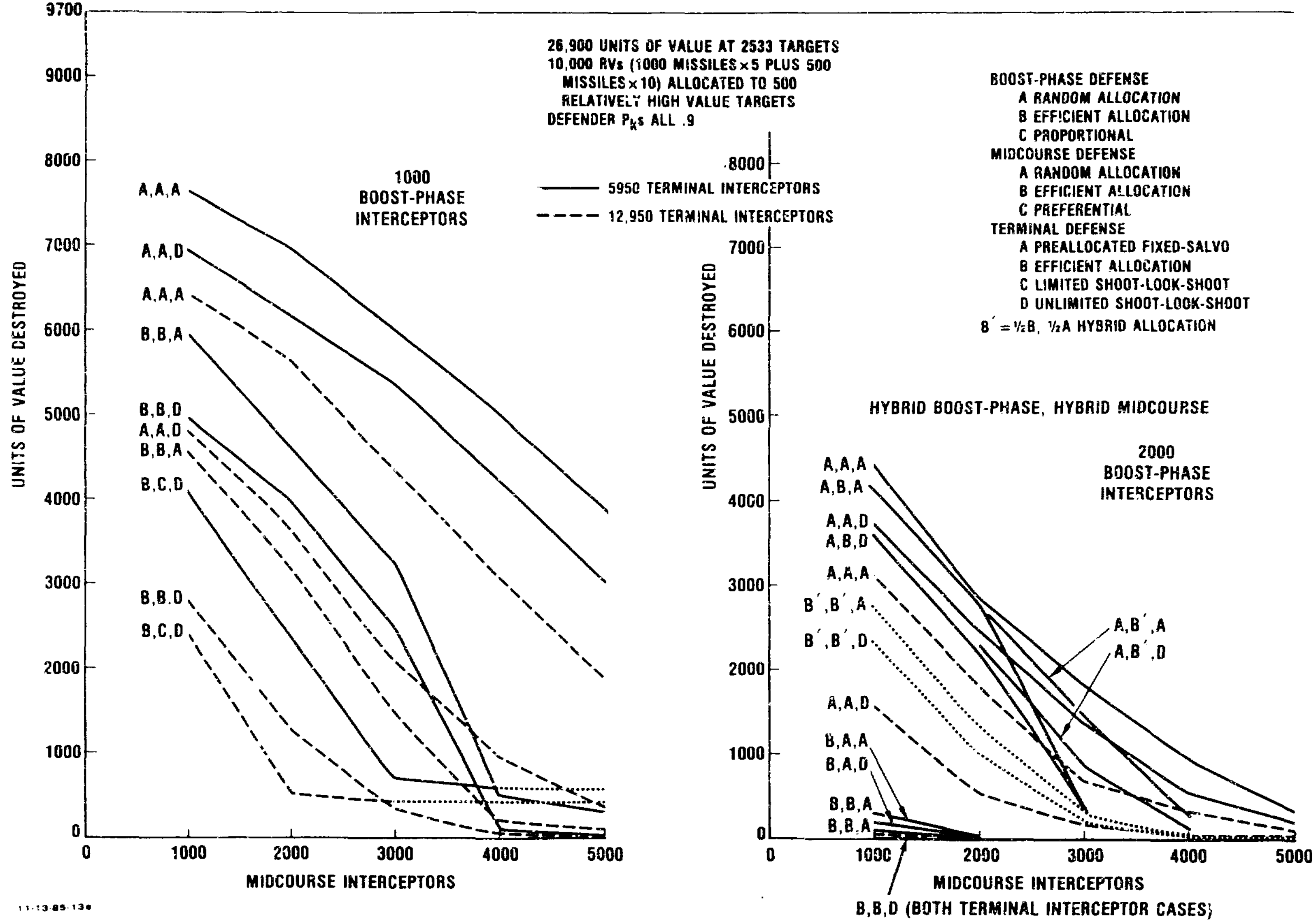

Figure 6. HYBRID BOOST-PHASE, HYBRID MIDCOURSE

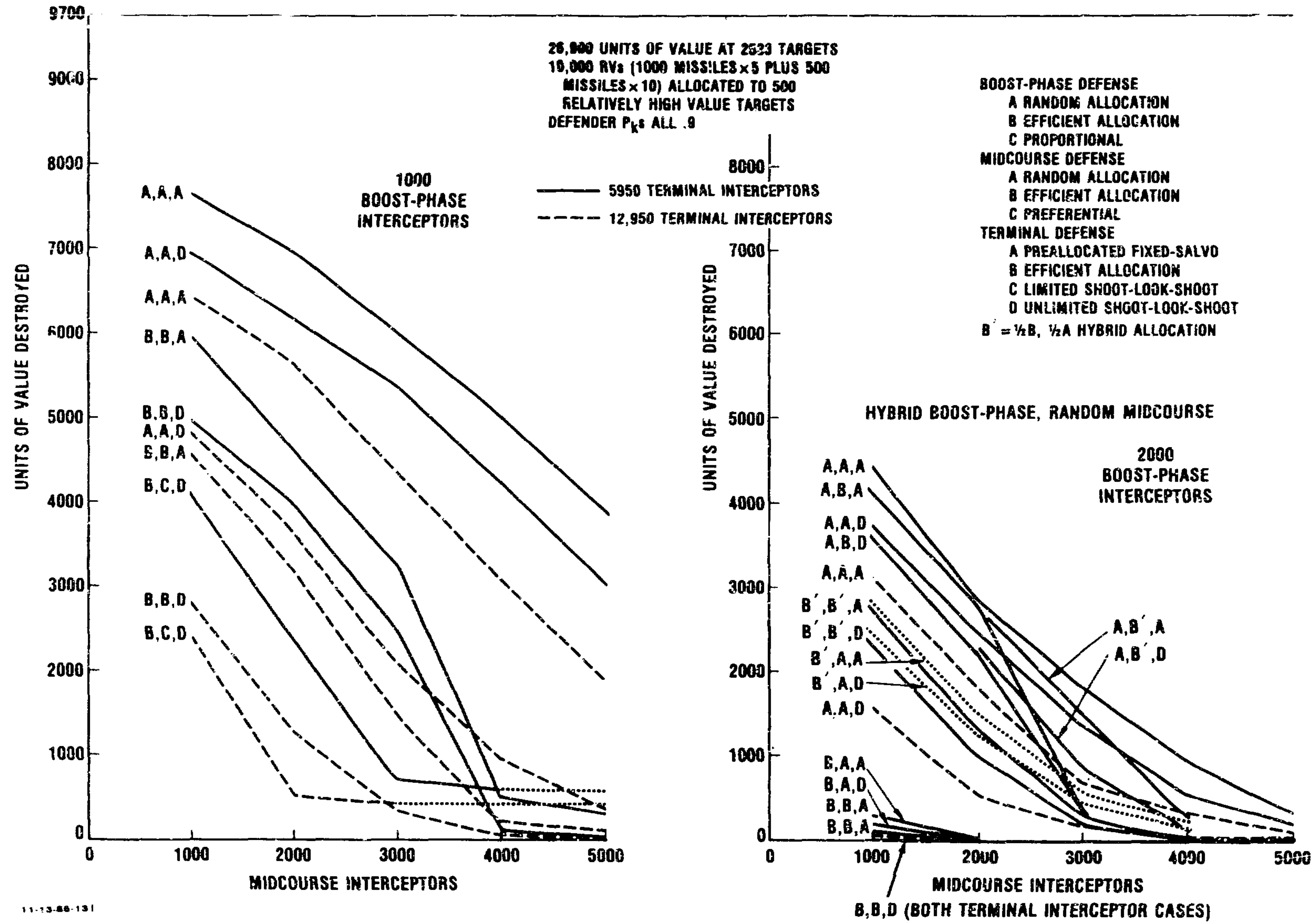

Figure 7. HYBRID BOOST-PHASE, RANDOM MIDCOURSE

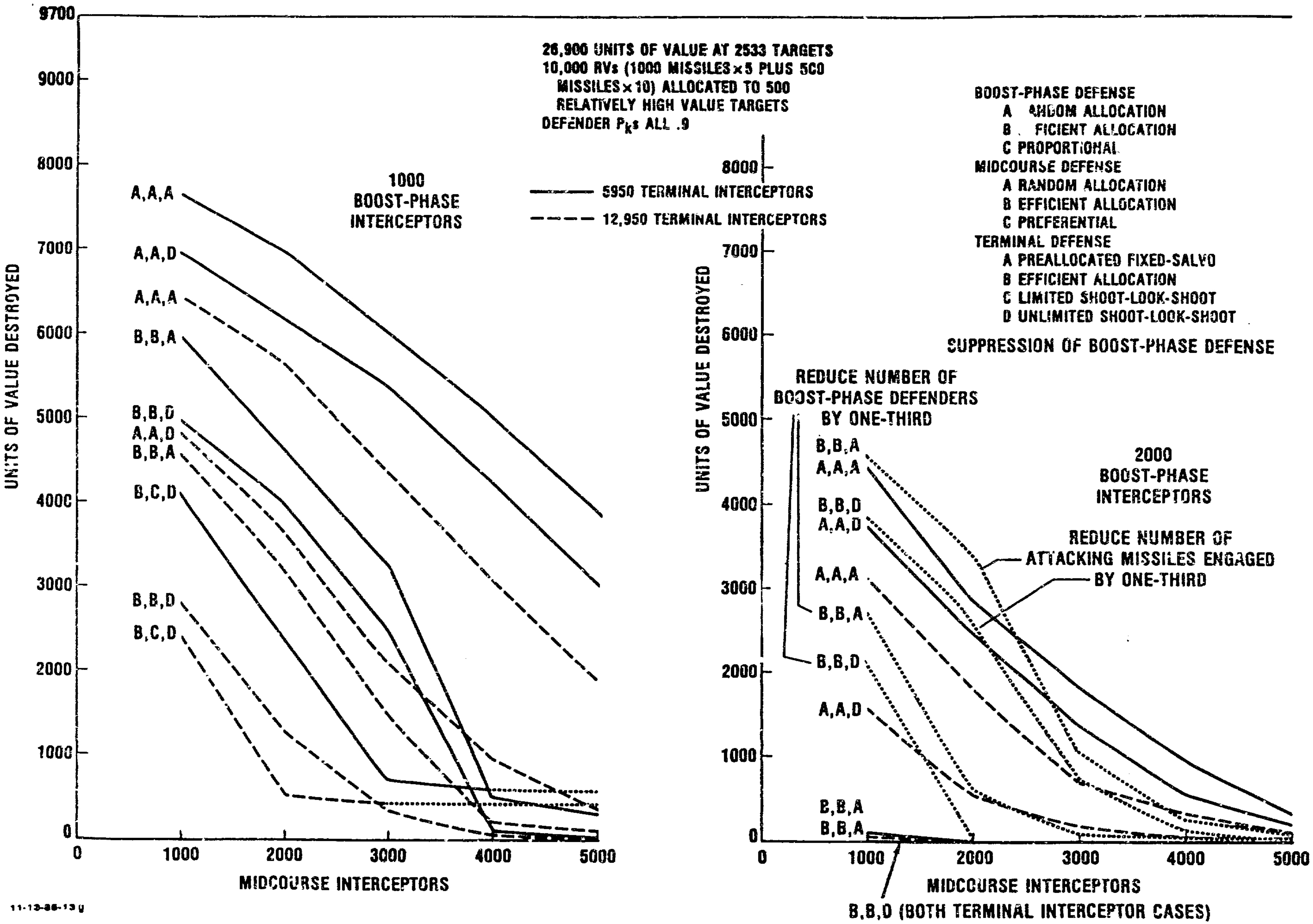

Figure 8. SUPPRESSION OF BOOST-PHASE DEFENSE

number of efficient midcourse defenders is sufficient to engage all of the surviving RVs, units of value destroyed is reduced to a small amount. Again, the behavior of results is the same as in previous examples with sufficient numbers of midcourse defenders.

In the second type of suppression one-third of the missiles, at random, are not engaged. The number of efficient midcourse defenders required to result in small damage is larger than in the first case.

12. Preferential Targeting by Boost-Phase Defenders

The left side of Figure 9 presents results for boost-phase defensive firing proportional to number of RVs on the missiles being attacked.

Two cases, C,B,A and C,B,D, are presented. They can be compared with B,B,A and B,B,D, and are seen to have less damage.

The reason why the return to the defender is not greater for proportional targeting of boost-phase defenders is that the attacking missiles have either 5 or 10 RVs each. If tthe mix of RVs were from 1 to 20 per missile, or something quite skewed, the improvement for the defender would be much greater.

13. Effects of Decoys

The left side of Figure 10 shows the effects of three decoy situations.

To begin, replace two of each five RVs by four heavy decoys, resulting in a total attack consisting of 6000 RVs and 8000 decoys. In all cases, assume that the BM/C^3 system of the defense is B,B,A.

In the first example, K = 2.5, which can be interpreted as meaning that the inherent error of the measuring device is such that RVs and decoys yield observations which are two and one-half standard deviations apart. A detection threshold is set resulting in PMD = .1 and PFA = .1 (probability of missed detection against an RV equal .1 and probability of false alarm against a decoy equal .1). In this case, the resulting units of value destroyed as a function of number of midcourse interceptors is lower than the baseline curve B,B,A. So this is not a good attacker option.

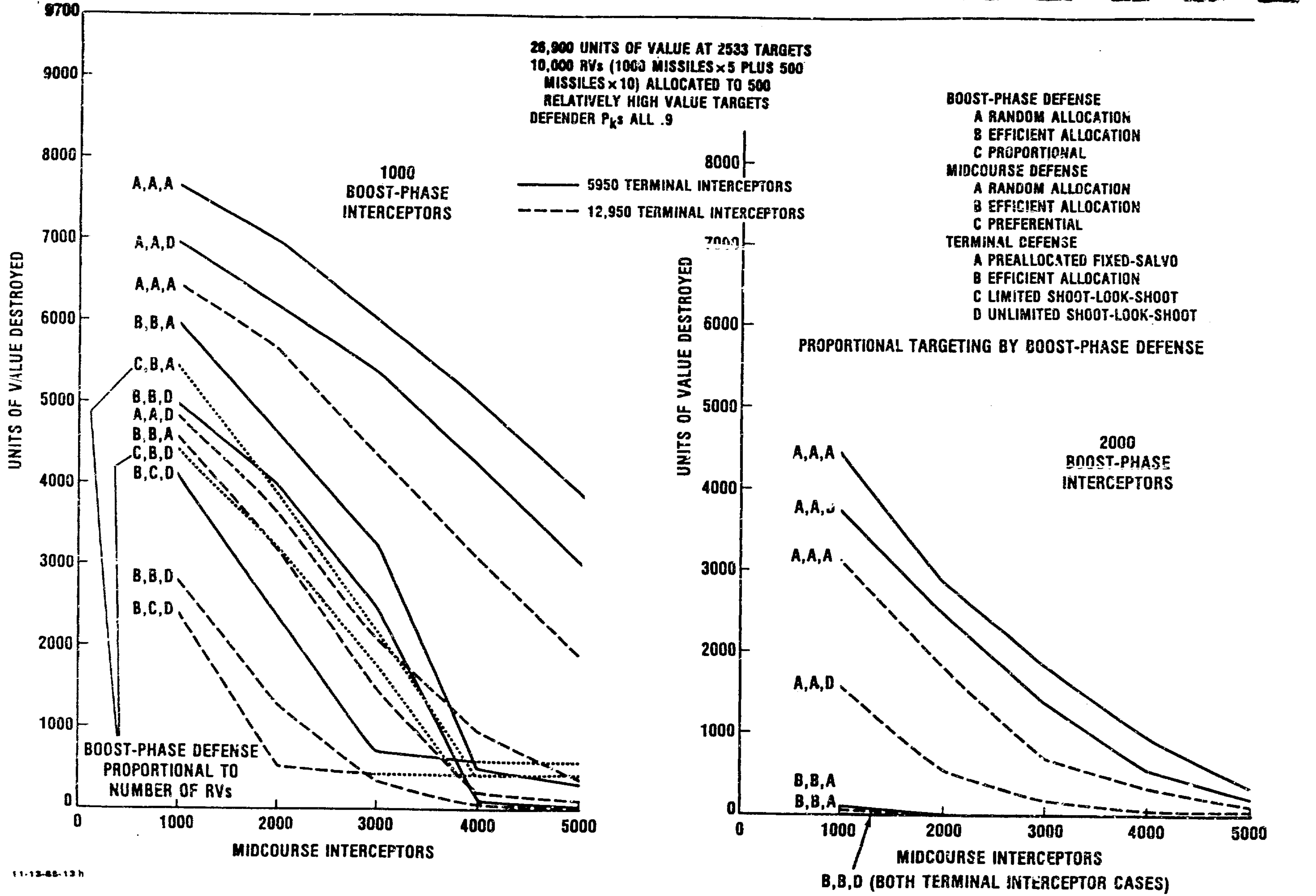

Figure 9: PROPORTIONAL TARGETING BY BOOST-PHASE DEFENSE

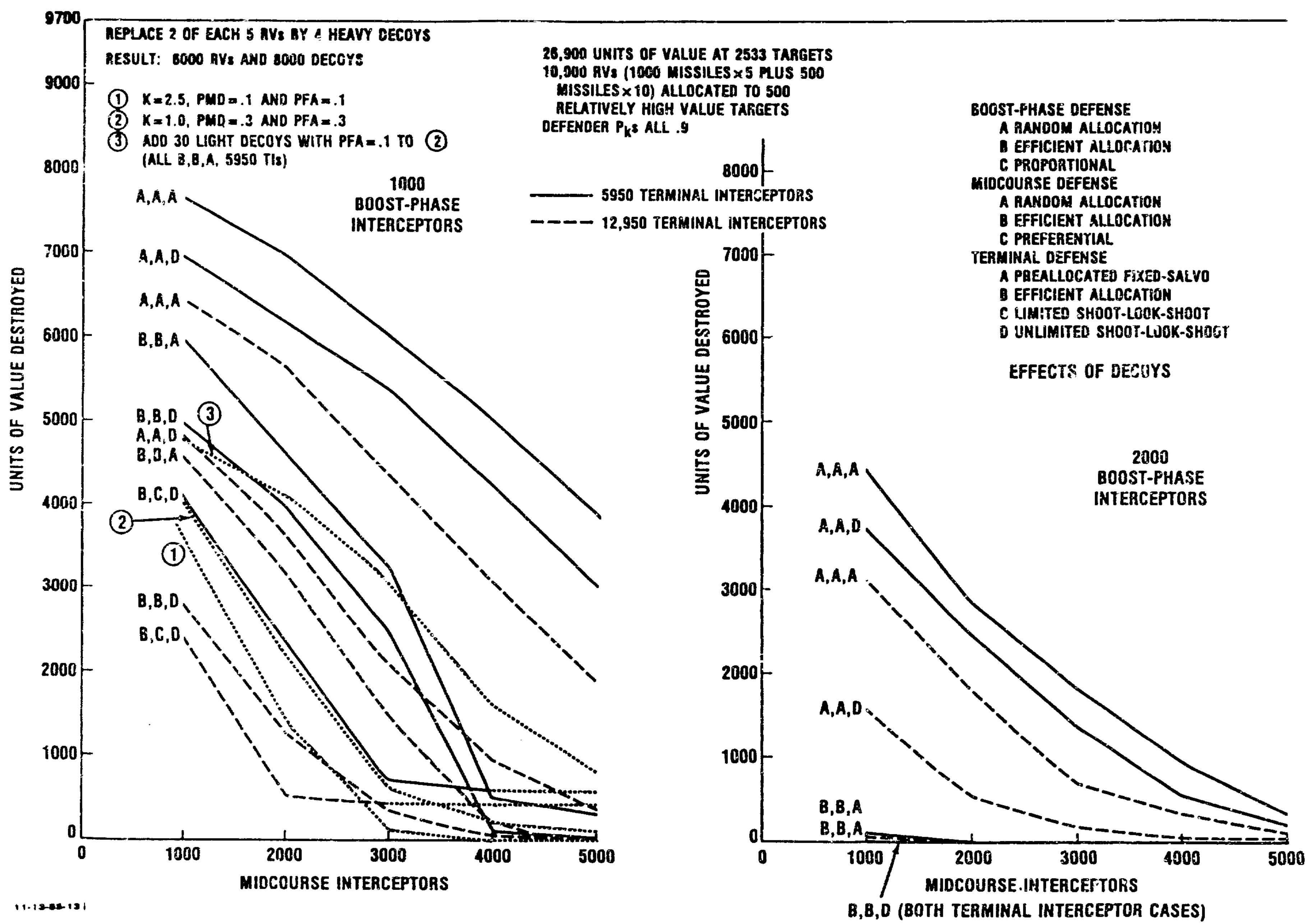

Figure 10. EFFECTS OF DECOYS

In the second example, K = 1.0, and the detection threshold yields PMD = .3 and PFA = .3. This is much worse for the defense than K = 2.5. But, still, it is not a good attacker option.

In the third example, however, assume that 30 light decoys can be added with PFA = .1 (smaller than the PFA = .3 for heavy decoys). The damage is smaller than B,B,A until the number of midcourse interceptors is greater than 3000, but then it is not reduced to near-zero. Even with 5000 interceptors, there is significant damage.

The third example is a striking demonstration of the difficulty in to the defense of dealing with decoys. Near-zero damage becomes very difficult to attain.

14. Concentrated Attack

This example is very different from those previously discussed, for in all of those cases the attack was held constant. Now, the effects of concentrating on a smaller set of targets are investigated.

Allocate all of the missiles to the 50 targets 24 through 73. The maximum possible amount of value destroyed is 1000 units as compared to 9700 units previously.

Both the left and right sides of Figure 11 show results of the new attack. On the left side the BM/C^3 assumption is B,B,D while on the right side it is A,B,D.

On the left side, for the original attack, 4000 interceptors yields near-zero damage while for the near attack even 6000 interceptors result in significant damage. This is because there are 200 RVs initially aimed at each target, and the process is sufficiently "lumpy" that at a few targets there are enough RVs remaining to exhaust the terminal defense.

On the right side, for the original attack 3000 efficient interceptors were enough to significantly blunt the attack. Here 4000 are required.

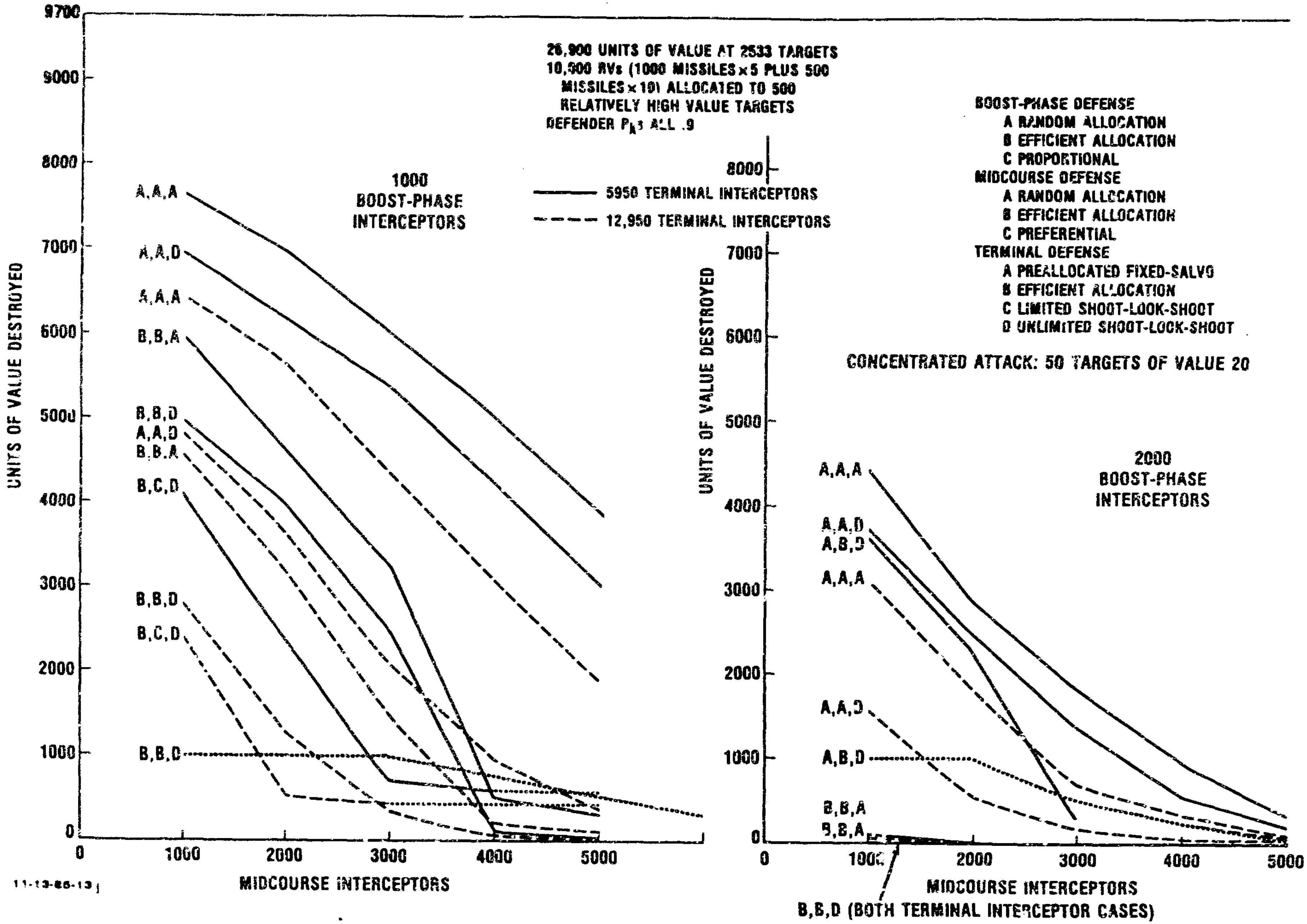

Figure 11. CONCENTRATED ATTACK

15. "Expected Value" Versus Monte Carlo Results

It is desirable to compare the results of "expected value" calculations with Monte Carlo calculations in order to illustrate the large differences which can occur under certain circumstances. This is done below for two cases.

In both cases the numbers of defenders are as follows:

Boost-Phase	1500
Midcourse	2000
Terminal	5950

The expected value calculation procedure follows. Assume that 1500 efficiently allocated boost-phase defenders confront 1500 missiles, and the P_K is .8. The expected number of missiles surviving is 300, comprised of 200 missiles with 5 RVs and 100 missiles with 10 RVs, for a total of 2000 RVs. Assume that 2000 efficiently allocated midcourse defenders confront the 2000 RVs. The expected number of survivors is 400.

Now assume that 400 RVs are uniformly distributed over the 500 targets. Since there are 470 targets of value 20 and 30 targets of value 10 assume that 376 of the first type and 24 of the second type are attacked by one RV each. With preallocated fixed-salvo defense of 2 interceptors, the probability of penetration is .04, so 376 x 20 x .04 = 300.8 units of value are killed at the higher-value targets. Similarly, 24 x 10 x .2 = 48 units of value are killed at the lower-value targets. The total is 348.8. With efficiently allocated terminal defense the equivalent figures are 376 x 20 x .00032 = 2.406 and 24 x 10 x .04 = 9.6. The total is 12.006.

Running the Monte Carlo model for 30 trials yields, for the two cases above, the following results:

	Mean of Sample	Standard Deviation of Sample
Preallocated Terminal Defense	1184	286
Efficient Terminal Defense	844	344

The inference about the process from the sample is as follows:

	Estimate of Population Mean	Standard Deviation of Estimate of Population Mean
Preallocated Terminal Defense	1184	53
Efficient Terminal Defense	844	64

Summarizing the expected Monte Carlo value calculations the following is obtained:

	Estimate of Mean with Expected Value Calculation	Estimate of Mean with Monte Carlo Calculation	Standard Deviation of Estimate of Mean with Monte Carlo Calculation
Preallocated Terminal Defense	349	1184	28
Efficient Terminal Defense	12	844	64

Performing exactly the same analysis with probability of kill equal .9 yields the following (40 Monte Carlo trials):

	Estimate of Mean with Expected Value Calculation	Estimate of Mean with Monte Carlo Calculation	Standard Deviation of Estimate of Mean with Monte Carlo Calculation
Preallocated Terminal Defense	25	246	28
Efficient Terminal Defense	1	126	20

Figure 12 and 13 show the calculations leading to the above results.

BOOST-PHASE DEFENSE: 1500 DEFENDERS WITH $P_k = .8$

MIDCOURSE DEFENSE: 2000 DEFENDERS WITH $P_k = .8$

TERMINAL DEFENSE: 5950 DEFENDERS WITH $P_k = .8$

ATTACK OF $1000 \times 5 + 500 \times 10 = 10{,}000$ RVs UNIFORMLY ALLOCATED TO 500 TARGETS

DETERMINISTIC

EXPECTED NUMBER OF RVs AFTER BOOST-PHASE DEFENSE: $200 \times 5 + 100 \times 10 = 2000$

EXPECTED NUMBER OF RVs AFTER MIDCOURSE DEFENSE: 400

EXPECTED DAMAGE ASSUMING UNIFORM ALLOCATION OF 400 RVs TO 400 TARGETS:

PREALLOCATED TERMINAL DEFENSE WITH $P_k = .8$: $[376 \times 20 \times .04 = 300.8] + [24 \times 10 \times .2 = 48] \cong 349$

EFFICIENT TERMINAL DEFENSE WITH $P_k = .8$: $[376 \times 20 \times .00032 = 2.406] + [24 \times 10 \times .04 = 9.6] \cong 12$

MONTE CARLO (30 TRIALS)

EXPECTED DAMAGE:

	SAMPLE MEAN $\bar{x}$ AND ESTIMATE OF POPULATION MEAN μ	SAMPLE STANDARD DEVIATION $\sigma(x)$	STANDARD DEVIATION OF ESTIMATE OF POPULATION MEAN $\sigma(\mu) = \sigma(x)/\sqrt{n-1}$
PREALLOCATED:	1184	286	53
EFFICIENT:	844	344	64

4-10-86-1

Figure 12. COMPARISON OF "EXPECTED VALUE" AND MONTE CARLO RESULTS FOR $P_K = .8$

BOOST-PHASE DEFENSE: 1500 DEFENDERS WITH $P_k = .9$

MIDCOURSE DEFENSE: 1000 DEFENDERS WITH $P_k = .9$

TERMINAL DEFENSE: 5950 DEFENDERS WITH $P_k = .9$

ATTACK OF $1000 \times 5 + 500 \times 10 = 10{,}000$ RVs UNIFORMLY ALLOCATED TO 500 TARGETS

DETERMINISTIC

EXPECTED NUMBER OF RVs AFTER BOOST-PHASE DEFENSE: $100 \times 5 + 50 \times 10 = 1000$

EXPECTED NUMBER OF RVs AFTER MIDCOURSE DEFENSE: 100

EXPECTED DAMAGE ASSUMING UNIFORM ALLOCATION OF 100 RVs TO 100 TARGETS:

PREALLOCATED TERMINAL DEFENSE WITH $P_k = .9$: $[94 \times 20 \times .01 = 18.8] + [6 \times 10 \times ,1 = 6.0] \cong 25$

EFFICIENT TERMINAL DEFENSE WITH $P_k = .9$: $[94 \times 20 \times .00001 = 0.018] + [6 \times 10 \times .01 = .6] \cong .6$

MONTE CARLO

EXPECTED DAMAGE:

	20 TRIALS			40 TRIALS (DIFFERENT SEED)		
	MEAN	STANDARD DEVIATIONS		MEAN	STANDARD DEVIATIONS	
	$\bar{x}$ AND μ	$\sigma(x)$	$\sigma(\mu)$	$\bar{x}$ AND μ	$\sigma(x)$	$\sigma(\mu)$
PREALLOCATED:	258	193	44	246	177	28
EFFICIENT:	143	116	27	126	127	20

4-10-86-2

Figure 13. COMPARISON OF "EXPECTED VALUE" AND MONTE CARLO RESULTS FOR $P_K = .9$

APPENDIX A

DEFINITIONS OF VARIABLES

DEFINITIONS OF VARIABLES

Name	Definition
AVES	Average value sum surviving, completed cumulatively as of each trial. Computed and used in MAIN.
AVED	Average value sum destroyed, computed cumulatively as of each trial. Computed and used in MAIN.
FIAWT	Sum of total number of RVs in array IAW(I,J). Computed and used in Subroutine BPD.
FKLD	Fraction of boost-phase defenders killed in boost-phase defense suppression. Set by user at beginning of Subroutine BPD, if desired.
FTARS	Fraction of missile targets which cannot be attacked by boost-phase defenders. Set by user at beginning of Subroutine BPD, if desired.
IAD(I,J,K)	Number of decoys of type K in the Jth group of RVs carried by missile I. In COMMON.
IAT(I,J)	Target aimed at by all of the RVs in the Jth group of RVs carried by missile I. In COMMON.
IAW(I,J)	Number of RVs in the Jth group of RVs carried by missile I. In COMMON.
IAWMT	Sum of total number of warheads on a particular missile. Computed and used in Subroutine BPD.
IAWS	Sum of total number of RVs in array IAW(I,J). Computed and used in MAIN.
IAWT	Sum of total number of RVs in array IAW(I,J). Computed and used in Subroutine BPD.
IAWT(I,J)	Temporary array of number of RVs in the Jth group of RVs carried by missile I. Computed and used in Subroutines MCD, MCD3, MCDWD.
IBPDS	Indicator for boost-phase defense suppression. If IBPDS=0, no boost-phase suppression. If IBPDS=1,2,3 or 4,boost-phase suppression of different types. Set by user at beginning of Subroutine BPD.

IBPDST	Working variable indicator for boost-phase defense suppression. May be set at 0 or 1. Set and used within Subroutine BPD.
IBPM	Indicator for hybrid boost-phase defense. If IBPM=0, no hybrid. If IBPM=1, hybrid. Used in MAIN.
ICMPLT	Indicator denoting whether boostiers receive complete coverage (i.e., whether all boosters necessarily receive at least one shot if possible). If ICMPLT=0, no complete coverage. If ICMPLT=1, complete coverage. Computed and used in Subroutine BPD.
IFRAC	Indicator for fraction of defense which is efficient or random. IFRAC=1 denotes efficient portion and IFRAC=2 denotes random portion. Used in MAIN.
IMCM	Indicator for hybrid midcourse defense. If IMCM=0, no hybrid. If IMCM=1, hybrid. Used in MAIN.
INDI(I,J)	Indicator for the Ith target and the Jth missile aimed at that target, giving the number of the missile. Computed and used in Subroutine MCD3.
INDJ(I,J)	Indicator for the Ith target and the Jth missile aimed at that target, giving the value of J. Computed and used in Subroutine MCD3.
INDTA(I)	Indicator for whether or not Ith missile has been assigned to a boost-phase defense shooter. If INDTA(I)=0, it has not. If INDTA(I)=1, it has. Computed and used in Subroutine BPD.
IRS	Seed of the random number generator, changed each time a random number is generated. In COMMON.
ITD(I)	Number of terminal defenders at target I. In COMMON.
ITDS	Sum of total number of terminal defenders. Computed and used in MAIN.
IV(I)	Value of target I. In COMMON.
IVS	Sum of total number of units of value. Computed and used in MAIN.
IVSD	Value sum destroyed. Computed and used in MAIN.
IVSS	Value sum surviving. Computed in Subroutine VALSURV. Used in MAIN.

JPR(I)	Number of shots taken by preallocated preferential defense with total number of interceptors I. Specified and used in Subroutine TD.
KPR(I,J)	Number of interceptors fired at the Jth incoming RV when the total number of interceptors is I. in a preallocated preferential defense. Specified and used in Subroutine TD.
LPI	Long printout indicator. If LPI=0, no long printout. If LPI=1, long printout. In COMMON.
NKLD	Number of boost-phase defenders killed in boost-phase defense suppression. Set by user at beginning of Subroutine BPD, if desired.
NOBJ1	Number of objects of type 1, namely RVs. Computed and used in Subroutine MCDWD.
NOBJ2	Number of objects of type 2, namely heavy decoys. Computed and used in Subroutine MCDWD.
NOBJ3	Number of objects of type 3, namely light decoys. Computed and used in Subroutine MCDWD.
NOBJT	Number of objects total. Computed and used in Subroutine MCDWD.
NTARS	Number of missile targets which cannot be attacked by boost-phase defenders. Set by user at beginning of Subroutine BPD, if desired.
NVT	Number of value targets. Created in VALUES. Passed in calling sequences of subroutines.
NW(I)	Number of RVs headed for target I. In COMMON.
NWRF(I)	Frequency distribution of number of targets receiving I penetrators at the end of the simulation. In COMMON.
NWRMP	Maximum number of RVs which can hit a target, plus 1. In COMMON.
NWT(I)	Temporary array of number of RVs headed for target I. Computed and used in Subroutine MCD3.
PFA1	Probability of false alarm against heavy decoys, where midcourse defender mistakenly calls a decoy an RV. Specified in Subroutine MCD and used in Subroutine MCDWD.

PFA2	Probability of false alarm against light decoys, where midcourse defender mistakenly calls a decoy an RV. Specified in Subroutine MCD and used in Subroutine MCDWD.
PMD	Probability of missed detection, where midcourse defender does not detect an RV. Specified in Subroutine MCD and used in Subroutine MCDWD.
RAT	Ratio of attacking RVs to terminal interceptors. Computed and used in Subroutine MCD3.
RATSL	Lower bound on ratio of attacking RVs to terminal interceptors for particular iteration of preferential defense algorithm. Computed and used in Subroutine MCD3.
RATSU	Upper bound on ratio of attacking RVs to terminal interceptors for particular iteration of preferential defense algorithm. Computed and used in Subroutine MCD3.
SD	Standard deviation of value sum surviving, computed cumulatively as of each trial. Computed and used in MAIN.
VSS(I)	Working variable to preserve value surviving at the end of the Ith trial of the simulation. Used in MAIN.

APPENDIX B

LISTING OF COMPUTER PROGRAM

```
      PROGRAM MAIN
1     FORMAT(1H1)
5     FORMAT(1H )
10    FORMAT(I10)
15    FORMAT(I20)
20    FORMAT(10I10)
30    FORMAT(10I10)
40    FORMAT(10F10.0)
50    FORMAT(25I5)
99    FORMAT(F20.2)
110   FORMAT(1H1,'NVT,IV,IVS')
120   FORMAT(1H1,'ITD,ITDS')
126   FORMAT(1H1,'AFTER ATTALL')
127   FORMAT(1H1,'AFTER BPD')
128   FORMAT(1H1,'AFTER MCD')
131   FORMAT(1H1,'IAT(I,1)')
132   FORMAT(1H0,'IAT(I,2)')
136   FORMAT(1H1,'IAW(I,1)')
137   FORMAT(1H0,'IAW(I,2)')
1381  FORMAT(1H1,'IAD(I,1,1)')
1382  FORMAT(1H0,'IAD(I,1,2)')
1383  FORMAT(1H1,'IAD(I,2,1)')
1384  FORMAT(1H0,'IAD(I,2,2)')
139   FORMAT(1H1,'NW AFTER TD')
C
200   FORMAT(1H0,'IVS')
201   FORMAT(1H ,'IMAIN')
202   FORMAT(1H ,'ITDS')
203   FORMAT(1H ,'IAWS')
2041  FORMAT(1H ,'NSHT OF BPD, IND')
2042  FORMAT(1H ,'IAWS AFTER BPD')
2051  FORMAT(1H ,'NSHT OF MCD, IND')
2052  FORMAT(1H ,'IAWS AFTER MCD')
2061  FORMAT(1H ,'IND OF TD')
2062  FORMAT(1H ,'NWS AFTER TD')
2063  FORMAT(1H ,'NWRF')
207   FORMAT(1H ,'IVSS,IVSD')
208   FORMAT(1H ,'AVES,AVED,SD')
C
      COMMON/VAL/IV(2533)
      COMMON/DEF/ITD(2533)
      COMMON/ATT/IAT(1500,2),IAW(1500,2)
      COMMON/DEC/IAD(1500,2,2)
      COMMON/RNSEED/IRS
      COMMON/TERM/NW(2533),NWRF(201),NWRMP
      COMMON/INDPRINT/LPI
      DIMENSION VSS(50)
C
      REAL TIME
C
      OPEN(6,FILE='OUT.DAT',STATUS='NEW')
      OPEN(7,FILE='SUP.DAT',STATUS='NEW')
C
      DO 9999 I1=1000,1000
      DO 9999 I2=3000,3000
      DO 9999 I3=1,2
      WRITE(7,1)
```

```
      IF(I3-2)410,420,401
 401  IF(I3-4)430,440,450
 410  I31=2
      IBPM=0
      I32=2
      IMCM=0
      I33=1
      GO TO 490
 420  I31=2
      IBPM=0
      I32=2
      IMCM=0
      I33=4
      GO TO 490
 430  I31=2
      IBPM=0
      I32=2
      IMCM=0
      I33=1
      GO TO 490
 440  I31=2
      IBPM=0
      I32=2
      IMCM=0
      I33=4
      GO TO 490
 450  I31=2
      IBPM=0
      I32=3
      IMCM=0
      I33=4
      GO TO 490
C
C --- LONG PRINT INDICATOR, LPI=0 NO, LPI=1 YES
C
 490  LPI=0
C
C --- RANDOM NUMBER SEED
C
      IRS=1
C
C --- SET UP TARGET VALUES
C --- SUBROUTINE VALUES CREATES ARRAY IV AND RETURNS NVT
C --- COMPUTE AND PRINT VALUE SUM IVS
C
      CALL VALUES(NVT)
C
      IF(LPI-1)510,500,500
 500  WRITE(6,110)
      WRITE(6,10)NVT
      WRITE(6,20)(IV(I),I=1,NVT)
 510  IVS=0
      DO 515 I=1,NVT
 515  IVS=IVS+IV(I)
      WRITE(6,200)
      WRITE(7,200)
      WRITE(*,200)
```

```
      WRITE(6,10)IVS
      WRITE(7,10)IVS
      WRITE(*,10)IVS
C
C --- MAIN LOOP ON NUMBER OF ITERATIONS (MAXIMUM=50)
C
      NUMB=1
      DO 8000 IMAIN=1,NUMB
      WRITE(6,201)
      WRITE(7,201)
      WRITE(*,201)
      WRITE(6,10)IMAIN
      WRITE(7,10)IMAIN
      WRITE(*,10)IMAIN
      CALL GETCPU(TIME)
      WRITE(*,99)TIME
C
C --- SET UP TERMINAL DEFENSE INVENTORY
C --- SUBROUTINE TDINV CREATES ARRAY ITD
C --- COMPUTE AND PRINT TERMINAL DEFENDERS SUM ITDS
C
      CALL TDINV(NVT)
C
      IF(LPI-1)610,600,600
  600 WRITE(6,120)
      WRITE(6,20)(ITD(I),I=1,NVT)
  610 ITDS=0
      DO 620 I=1,NVT
  620 ITDS=ITDS+ITD(I)
      WRITE(6,202)
      WRITE(7,202)
      WRITE(*,202)
      WRITE(6,10)ITDS
      WRITE(7,10)ITDS
      WRITE(*,10)ITDS
C
C --- SET UP ATTACK ALLOCATION
C --- SUBROUTINE ATTALL RETURNS NMIS AND MNWT AND CREATES ARRAYS IAT AND IAW
C --- COMPUTE AND PRINT WARHEADS SUM IAWS
C
      CALL ATTALL(NMIS,MNWT)
C
      IF(LPI-1)1075,1050,1050
 1050 WRITE(6,126)
      WRITE(6,131)
      WRITE(6,30)(IAT(I,1),I=1,NMIS)
      WRITE(6,132)
      WRITE(6,30)(IAT(I,2),I=1,NMIS)
      WRITE(6,136)
      WRITE(6,30)(IAW(I,1),I=1,NMIS)
      WRITE(6,137)
      WRITE(6,30)(IAW(I,2),I=1,NMIS)
      WRITE(6,1381)
      WRITE(6,30)(IAD(I,1,1),I=1,NMIS)
      WRITE(6,1382)
      WRITE(6,30)(IAD(I,1,2),I=1,NMIS)
      WRITE(6,1383)
```

```
      WRITE(6,30)(IAD(I,2,1),I=1,NMIS)
      WRITE(6,1384)
      WRITE(6,30)(IAD(I,2,2),I=1,NMIS)
1075  IAWS=0
      DO 1100 I=1,NMIS
1100  IAWS=IAWS+IAW(I,1)+IAW(I,2)
      WRITE(6,203)
      WRITE(7,203)
      WRITE(*,203)
      WRITE(6,10)IAWS
      WRITE(7,10)IAWS
      WRITE(*,10)IAWS
C
C --- BOOST PHASE DEFENSE
C --- IND=1,RANDOM
C --- IND=2,EFFICIENT
C --- IND=3,PROPORTIONAL TO NUMBER OF RVS
C
      IF(IBPH-1)1150,1110,1110
C
1110  NTAR=NMIS
      PK=.9
      NSHT=.5*I1
      IND=2
      IFRAC=1
      GO TO 1160
1130  NSHT=.5*I1
      IND=1
      IFRAC=2
      GO TO 1160
C
1150  NSHT=I1
      NTAR=NMIS
      PK=.9
      IND=I31
C
1160  WRITE(7,2041)
      WRITE(*,2041)
      WRITE(7,20)NSHT,IND
      WRITE(*,20)NSHT,IND
      CALL BPD(NSHT,NTAR,MNWT,PK,IND)
C
      IF(LPT-1)1195,1190,1190
1190  WRITE(6,127)
      WRITE(6,131)
      WRITE(6,30)(IAT(I,1),I=1,NMIS)
      WRITE(6,132)
      WRITE(6,30)(IAT(I,2),I=1,NMIS)
      WRITE(6,136)
      WRITE(6,30)(IAW(I,1),I=1,NMIS)
      WRITE(6,137)
      WRITE(6,30)(IAW(I,2),I=1,NMIS)
      WRITE(6,1381)
      WRITE(6,30)(IAD(I,1,1),I=1,NMIS)
      WRITE(6,1382)
      WRITE(6,30)(IAD(I,1,2),I=1,NMIS)
      WRITE(6,1383)
```

```
        WRITE(6,30)(IAD(I,2,1),I=1,NMIS)
        WRITE(6,1384)
        WRITE(6,30)(IAD(I,2,2),I=1,NMIS)
 1195   IAWS=0
        DO 1200 I=1,NMIS
C
 1200   IAWS=IAWS+IAW(I,1)+IAW(I,2)
        WRITE(6,2042)
        WRITE(7,2042)
        WRITE(*,2042)
        WRITE(6,10)IAWS
        WRITE(7,10)IAWS
        WRITE(*,10)IAWS
C
        IF(IBPH-1)1208,1205,1205
 1205   IF(IFRAC-1)1130,1130,1208
C
C ---   MIDCOURSE DEFENSE
C ---   IND=1,RANDOM
C ---   IND=2,EFFICIENT
C ---   IND=3,PREFERENTIAL
C
 1208   IF(IMCM-1)1250,1210,1210
C
 1210   NTAR=NMIS
        PK=.9
        NSHT=.5*.2
        IND=2
        IFRAC=1
        GO TO 1260
 1230   NSHT=.5*12
        IND=1
        IFRAC=2
        GO TO 1260
C
 1250   NSHT=12
        NTAR=NMIS
        PK=.9
        IND=I32
C
 1260   WRITE(7,2051)
        WRITE(*,2051)
        WRITE(7,20)NSHT,IND
        WRITE(*,20)NSHT,IND
        CALL MCD(NVT,NSHT,NTAR,MNWT,PK,IND)
C
        IF(LPI-1)1295,1290,1290
 1290   WRITE(6,128)
        WRITE(6,131)
        WRITE(6,30)(IAT(I,1),I=1,NMIS)
        WRITE(6,132)
        WRITE(6,30)(IAT(I,2),I=1,NMIS)
        WRITE(6,136)
        WRITE(6,30)(IAW(I,1),I=1,NMIS)
        WRITE(6,137)
        WRITE(6,30)(IAW(I,2),I=1,NMIS)
        WRITE(6,1381)
```

```
            WRITE(6,30)(IAD(I,1,1),I=1,NMIS)
            WRITE(6,1382)
            WRITE(6,30)(IAD(I,1,2),I=1,NMIS)
            WRITE(6,1383)
            WRITE(6,30)(IAD(I,2,1),I=1,NMIS)
            WRITE(6,1384)
            WRITE(6,30)(IAD(I,2,2),I=1,NMIS)
C
      1295  IAWS=0
            DO 1300 I=1,NMIS
      1300  IAWS=IAWS+IAW(I,1)+IAW(I,2)
            WRITE(6,2052)
            WRITE(7,2052)
            WRITE(*,2052)
            WRITE(6,10)IAWS
            WRITE(7,10)IAWS
            WRITE(*,10)IAWS
C
            IF(IMCM-1)1308,1305,1305
      1305  IF(IFRAC-1)1230,1230,1308
C
C ---       TERMINAL DEFENSE
C ---       SUBROUTINE TD RETURNS NUMBER OF WEAPONS ON TARGETS
C ---       AFTER TERMINAL DEFENSE: ARRAY NW
C ---       IND=1,PREALLOCATED FIXED-SALVO
C ---       IND=2,EFFICIENT
C ---       IND=3,SHOOT-LOOK-SHOOT LIMITED
C ---       IND=4,SHOOT-LOOK-SHOOT UNLIMITED
C
      1308  PK=.9
            IND=133
            WRITE(7,2061)
            WRITE(*,2061)
            WRITE(7,15)IND
            WRITE(*,15)IND
            CALL TD(NVT,NMIS,MNWT,PK,IND)
C
            IF(LPI-1)1395,1390,1390
      1390  WRITE(6,139)
            WRITE(6,20)(NW(I),I=1,NVT)
C
      1395  NWS=0
            DO 1400 I=1,NVT
      1400  NWS=NWS+NW(I)
            WRITE(6,2062)
            WRITE(7,2062)
            WRITE(*,2062)
            WRITE(6,10)NWS
            WRITE(7,10)NWS
            WRITE(*,10)NWS
            WRITE(6,2063)
            WRITE(7,2063)
            WRITE(*,2063)
            WRITE(6,50)(NWRF(I),I=1,NWRMP)
            WRITE(7,50)(NWRF(I),I=1,NWRMP)
            WRITE(*,50)(NWRF(I),I=1,NWRMP)
C
```

```
C ---   VALUE SURVIVING
C ---   SUBROUTINE VALSURV RETURNS VALUE SUM SURVIVING: IVSS
C
        CALL VALSURV(NVT,IVSS)
        IVSD=IVS-IVSS
C
        WRITE(6,207)
        WRITE(7,207)
        WRITE(*,207)
        WRITE(6,20)IVSS,IVSD
        WRITE(7,20)IVSS,IVSD
        WRITE(*,20)IVSS,IVSD
C
C ---   PRESERVE RESULTS OF THIS TRIAL
C
        VSS(IMAIN)=IVSS
C
C ---   COMPUTE AVERAGE AND STANDARD DEVIATION
C
        FIMAIN=IMAIN
        FIVS=IVS
        VSST=0.
        DO 6100 I=1,IMAIN
 6100   VSST=VSST+VSS(I)
        AVES=VSST/FIMAIN
        AVED=FIVS-AVES
        SDIF2=0.
        DO 6200 I=1,IMAIN
 6200   SDIF2=SDIF2+(VSS(I)-AVES)**2
        SD=(SDIF2/FIMAIN)**.5
C
        WRITE(6,208)
        WRITE(7,208)
        WRITE(*,208)
        WRITE(6,40)AVES,AVED,SD
        WRITE(7,40)AVES,AVED,SD
        WRITE(*,40)AVES,AVED,SD
C
C ---   END OF MAIN LOOPS
C
 8000   CONTINUE
C
 9999   CONTINUE
        END
```

```
      SUBROUTINE VALUES(NVT)
      COMMON/VAL/IV(2533)
C
      NVT=2533
C
C --- VALUES
C
C --- SLBM 3(200)
C --- C3 10(100)
C --- ABM 10(50)
C --- BOMBER 50(20), AIR DEF 20(20), CONUS MIL 200(20), TH MIL 200(20)
C --- C3 40(10), CONUS CIV 500(10), TH CIV 500(10)
C --- ICBM 1000(5)
C
      DO 10 I=1,3
   10 IV(I)=200
      DO 20 I=4,13
   20 IV(I)=100
      DO 30 I=14,23
   30 IV(I)=50
      DO 40 I=24,493
   40 IV(I)=20
      DO 50 I=494,1533
   50 IV(I)=10
      DO 60 I=1534,2533
   60 IV(I)=5
C
      CONTINUE
      RETURN
      END
```

```
0001
0002              SUBROUTINE TDINV(NVT)
0003              COMMON/VAL/IV(2533)
0004              COMMON/DEF/ITD(2533)
0005      C
0006      C ---   TERMINAL DEFENSE INVENTORY
0007      C
0008              DO 500 I=1,NVT
0009       500    ITD(I)=IV(I)/4
0010      C
0011              CONTINUE
0012              RETURN
0013              END
```

PROGRAM SECTIONS

Name	Bytes	Attributes
0 $CODE	64	PIC CON REL LCL SHR EXE RD NOWRT LONG
2 $LOCAL	8	PIC CON REL LCL NOSHR NOEXE RD WRT LONG
3 VAL	10132	PIC OVR REL GBL SHR NOEXE RD WRT LONG
4 DEF	10132	PIC OVR REL GBL SHR NOEXE RD WRT LONG
Total Space Allocated	20336	

ENTRY POINTS

Address	Type	Name	References
0-00000000		TDINV	2#

VARIABLES

Address	Type	Name	Attributes	References	
2-00000000	I*4	I		8=	9(2)
AP-00000004@	I*4	NVT		2	8

ARRAYS

Address	Type	Name	Attributes	Bytes	Dimensions	References	
4-00000000	I*4	ITD	COMM	10132	(2533)	4	9=
3-00000000	I*4	IV	COMM	10132	(2533)	3	9

```
      SUBROUTINE ATTALL(NMIS,MNWT)
      COMMON/ATT/IAT(1500,2),IAW(1500,2)
      COMMON/DEC/IAD(1500,2,2)
      COMMON/TERM/NW(2533),NWRF(201),NWRMP
C
      NMIS=1500
      MNWT=2
C
C ---  MISSILE AND RV ASSIGNMENT
C ---  IASS=1 REGULAR 24-523  500 TARGETS,20 EACH
C ---  IASS=2 CONCENTRATED 24-73 50 TARGETS,200 EACH
C ---  IASS=3 REGULAR 24-523 PLUS DECOYS
C ---  NWRMP=MAXIMUM NUMBER OF RVS PER TARGET
C
      IASS=1
C
      IF(IASS-2)5,105,205
C
C ---  REGULAR 24-523
C
  5   DO 10 I=1,500
      IAT(I,1)=I+23
      IAW(I,1)=5
      IAD(I,1,1)=0
 10   IAD(I,1,2)=0
      DO 20 I=501,1000
      IAT(I,1)=I-477
      IAW(I,1)=5
      IAD(I,1,1)=0
 20   IAD(I,1,2)=0
      DO 30 I=1001,1500
      IAT(I,1)=I-977
      IAW(I,1)=10
      IAD(I,1,1)=0
 30   IAD(I,1,2)=0
      NWRMP=21
      GO TO 1000
C
C ---  CONCENTRATED 24-73
C
 105  DO 110 I=1,1000
      IT=(I-1)/20+1
      ITT=IT+23
      IAT(I,1)=ITT
      IAW(I,1)=5
      IAD(I,1,1)=0
 110  IAD(I,1,2)=0
      DO 120 I=1001,1500
      IP=I-1000
      IT=(IP-1)/10+1
      ITT=IT+23
      IAT(I,1)=ITT
      IAW(I,1)=10
      IAD(I,1,1)=0
 120  IAD(I,1,2)=0
      NWRMP=201
      GO TO 1000
```

```
0058      C
0059      C ---     REGULAR 24-523 PLUS DECOYS
0060      C
0061       205      DO 210 I=1,500
0062                IAT(I,1)=I+23
0063                IAW(I,1)=3
0064                IAD(I,1,1)=4
0065       210      IAD(I,1,2)=30
0066                DO 220 I=501,1000
0067                IAT(I,1)=I-477
0068                IAW(I,1)=3
0069                IAD(I,1,1)=4
0070       220      IAD(I,1,2)=30
0071                DO 230 I=1001,1500
0072                IAT(I,1)=I-977
0073                IAW(I,1)=6
0074                IAD(I,1,1)=8
0075       230      IAD(I,1,2)=30
0076                NWRMP=21
0077                GO TO 1000
0078      C
0079       1000     CONTINUE
0080                RETURN
0081                END
```

PROGRAM SECTIONS

Name	Bytes	Attributes
0 $CODE	918	PIC CON REL LCL SHR EXE RD NOWRT LONG
2 $LOCAL	20	PIC CON REL LCL NOSHR NOEXE RD WRT LONG
3 ATT	24000	PIC OVR REL GBL SHR NOEXE RD WRT LONG
4 DEC	24000	PIC OVR REL GBL SHR NOEXE RD WRT LONG
5 TERM	10940	PIC OVR REL GBL SHR NOEXE RD WRT LONG
Total Space Allocated	59878	

ENTRY POINTS

Address	Type	Name	References
0-00000000		ATTALL	1#

VARIABLES

Address	Type	Name	Attributes	References							
2-00000004	I*4	I		21=	22(2)	23	24	25	26=	27(2)	28
				29	30	31=	32(2)	33	34	35	41=
				42	44	45	46	47	48=	49	52
				53	54	55	61=	62(2)	63	64	65
				66=	67(2)	68	69	70	71=	72(2)	73

```
      SUBROUTINE BPD(NSHT,NTAR,MNWT,PK,IND)
      COMMON/ATT/IAT(1500,2),IAW(1500,2)
      COMMON/DEC/IAD(1500,2,2)
      COMMON/RNSEED/IRS
      DIMENSION INDTA(1500)
      DIMENSION IAWSV(1500,2)
C
      SHT=NSHT
      TAR=NTAR
C
C --- BOOST-PHASE DEFENSE SUPPRESSION
C --- IBPDS=0 NO DEFENSE SUPPRESSION
C --- IBPDS=1 REDUCE NUMBER OF SHOOTERS BY NKLD
C --- IBPDS=2 REDUCE NUMBER OF SHOOTERS BY NSHT*FKLD
C --- IBPDS=3 DO NOT ENGAGE NUMBER OF MISSILES NTARS
C --- IBPDS=4 DO NOT ENGAGE FRACTION OF MISSILES FTARS
C
      IBPDS=0
      NKLD=667
      FKLD=.333
      NTARS=500
      FTARS=.333
C
      IF(IBPDS-1)900,810,802
  802 IF(IBPDS-3)820,830,840
C
  810 IBPDST=0
      NSHT=NSHT-NKLD <SHT=NSHT
      GO TO 900
C
  820 IBPDST=0
      SHT=SHT-SHT*FKLD
      NSHT=SHT
      GO TO 900
C
  830 IBPDST=1
      TARS=NTARS
      FTARS=TARS/TAR
      GO TO 900
C
  840 IBPDST=1
      GO TO 900
C
C --- RANDOM, EFFICIENT OR PROPORTIONAL TO RVS
C
  900 IF(IND-2)1000,2000,3000
C
C --- SELECT TARGETS OF BPD AT RANDOM, KILL WITH PROBABILITY PK
C --- MAY ENGAGE AND KILL SAME TARGET SEVERAL TIMES
C --- BOOST-PHASE DEFENSE SUPPRESSION OPTIONS IBPDS=3 AND 4 SAVE FTARS
C --- LOWER-NUMBERED MISSILES
C
 1000 DO 1100 I=1,NSHT
      A=RAN(IRS)
      IF(IBPDST-1)1005,1002,1002
 1002 IF(A-FTARS)1100,1100,1005
 1005 AMIS=A*TAR
```

```
            IMIS=JNINT(AMIS)
            IMIS=MAX(IMIS,1)
            A=RAN(IRS)
            IF(A-PK)1010,1010,1100
      1010  DO 1020 J=1,MNWT
            IAT(IMIS,J)=0
            IAW(IMIS,J)=0
            IAD(IMIS,J,1)=0
      1020  IAD(IMIS,J,2)=0
      1100  CONTINUE
      C
            GO TO 4000
      C
      C --- SELECT TARGETS OF BPD EFFICIENTLY, KILL WITH PROBABILITY PK
      C --- BOOST-PHASE DEFENSE SUPPRESSION OPTIONS IBPDS=3 AND 4 SAVE FTARS
      C --- ANY-NUMBERED MISSILES
      C
      2000  IF(NSHT-NTAR)2100,2400,2600
      C
      C --- SHOOTERS FEWER THAN TARGETS
      C
      2100  DO 2101 I=1,NTAR
      2101  INDTA(I)=0
      C
            DO 2200 I=1,NSHT
            IF(IBPDST-1)2105,2102,2102
      2102  A=RAN(IRS)
            IF(A-FTARS)2200,2200,2105
      2105  A=RAN(IRS)
            AMIS=A*TAR
            IMIS=JNINT(AMIS)
            IMIS=MAX(IMIS,1)
            IF(INDTA(IMIS)-1)2152,2105,2105
      2152  INDTA(IMIS)=1
            A=RAN(IRS)
            IF(A-PK)2160,2160,2200
      2160  DO 2165 J=1,MNWT
            IAT(IMIS,J)=0
            IAW(IMIS,J)=0
            IAD(IMIS,J,1)=0
      2165  IAD(IMIS,J,2)=0
      2200  CONTINUE
      C
            GO TO 4000
      C
      C --- SHOOTERS EQUAL TO TARGETS
      C
      2400  CONTINUE
            DO 2500 I=1,NSHT
            IF(IBPDST-1)2405,2402,2402
      2402  A=RAN(IRS)
            IF(A-FTARS)2500,2500,2405
      2405  A=RAN(IRS)
            IF(A-PK)2450,2450,2500
      2450  DO 2455 J=1,MNWT
            IAT(I,J)=0
            IAW(I,J)=0
```

```
            IAD(I,J,1)=0
     2455   IAD(I,J,2)=0
     2500   CONTINUE
C
            GO TO 4000
C
C ---       SHOOTERS GREATER THAN TARGETS
C
     2600   DO 2605 I=1,NTAR
     2605   INDTA(I)=0
C
            R=SHT/TAR
            IR=R
            FIR=IR
            P1=R-FIR
            IRP1=IR+1
C
C ---       IRP1 TO P1 OF TARGETS
C ---       IR TO (1-P1) OF TARGETS
C ---       (EXACT ASSIGNMENT, NOT APPROXIMATE ASSIGNMENT AS IN MIDCOURSE.
C
            TP1=TAR*P1
            NTP1=JNINT(TP1)
C
C ---       TARGETS RECEIVING IRP1
C
            DO 2800 I=1,NTP1
            IF(IBPDST-1)2705,2702,2702
     2702   A=RAN(IRS)
            IF(A-FTARS)2800,2800,2705
     2705   A=RAN(IRS)
            AMIS=A*TAR
            IMIS=JNINT(AMIS)
            IMIS=MAX(IMIS,1)
            IF(INDTA(IMIS)-1)2760,2705,2705
     2760   INDTA(IMIS)=1
            DO 2780 J=1,IRP1
            A=RAN(IRS)
            IF(A-PK)2770,2770,2780
     2770   DO 2775 K=1,MNWT
            IAT(IMIS,K)=0
            IAW(IMIS,K)=0
            IAD(IMIS,K,1)=0
     2775   IAD(IMIS,K,2)=0
     2780   CONTINUE
     2800   CONTINUE
C
C ---       TARGETS RECEIVING IR
C
            DO 2900 I=1,NTAR
            IF(INDTA(I)-1)2810,2900,2900
     2810   IF(IBPDST-1)2815,2812,2812
     2812   A=RAN(IRS)
            IF(A-FTARS)2900,2900,2815
     2815   DO 2850 J=1,IR
            A=RAN(IRS)
            IF(A-PK)2820,2820,2850
```

```
 2820   DO 2825 K=1,MNWT
        IAT(I,K)=0
        IAW(I,K)=0
        IAD(IMIS,K,1)=0
 2825   IAD(IMIS,K,2)=0
 2850   CONTINUE
 2900   CONTINUE
C
        GO TO 4000
C
C ---   ALLOCATE SHOTS OF BPD TO MISSILES IN PROPORTION TO
C ---   NUMBER OF RVS
C
 3000   IAWT=0
        DO 3001 I=1,NTAR
        DO 3001 J=1,MNWT
 3001   IAWT=IAWT+IAW(I,J)
        FIAWT=IAWT
C
        IF(NSHT-NTAR)3002,3300,3300
C
C ---   SHOOTERS LESS THAN TARGETS
C
 3002   SPW=SHT/FIAWT
        DO 3003 I=1,NTAR
 3003   INDTA(I)=0
C
        ITEMP=0
 3005   IF(ITEMP-NSHT)3010,3010,3250
 3010   A=RAN(IRS)
        AMIS=A*TAR
        IMIS=JNINT(AMIS)
        IMIS=MAX(IMIS,1)
        IF(INDTA(IMIS)-0)3020,3020,3010
 3020   IAWMT=0
        DO 3030 J=1,MNWT
 3030   IAWMT=IAWMT+IAW(IMIS,J)
        FIAWMT=IAWMT
        SHTS=SPW*FIAWMT
        ISHTS=SHTS
        FISHTS=ISHTS
        REM=SHTS-FISHTS
        A=RAN(IRS)
        IF(A-REM)3040,3040,3065
 3040   ISHTS=ISHTS+1
 3045   ITEMP=ITEMP+ISHTS
        DO 3150 J=1,ISHTS
        A=RAN(IRS)
        IF(A-PK)3120,3120,3150
 3120   DO 3130 K=1,MNWT
        IAT(IMIS,K)=0
        IAW(IMIS,K)=0
        IAD(IMIS,K,1)=0
 3130   IAD(IMIS,K,2)=0
 3150   CONTINUE
        GO TO 3005
C
```

```
 3250  CONTINUE
       GO TO 4000
C
C ---  SHOOTERS MORE THAN TARGETS
C ---  ICMPLT=0(NOT COMPLETE COVERAGE) IF NOT AT LEAST ONE SHOT PER MISSILE
C ---  ICMPLT=1(COMPLETE COVERAGE) IF AT LEAST ONE SHOT PER MISSILE
C
 3300  ICMPLT=1
C
       IF(ICMPLT-1)3500,3700,3700
C
C ---  ICMPLT=0
C
 3500  SPW=SHT/FIAWT
       DO 3690 IMIS=1,NTAR
       IAWMT=0
       DO 3520 J=1,MNWT
 3520  IAWMT=IAWMT+1/W(IMIS,J)
       FIAWMT=IAWMT
       SHTS=SPW*FIAWMT
       ISHTS=SHTS
       FISHTS=ISHTS
       REM=SHTS-FISHTS
       A=RAN(IRS)
       IF(A-REM)3540,3540,3545
 3540  ISHTST=ISHTS+1
       GO TO 3600
 3545  ISHTST=ISHTS
C
 3600  DO 3650 J=1,ISHTST
       A=RAN(IRS)
       IF(A-PK)3620,3620,3650
 3620  DO 3630 K=1,MNWT
       IAT(IMIS,K)=0
       IAW(IMIS,K)=0
       IAD(IMIS,K,1)=0
 3630  IAD(IMIS,K,2)=0
 3650  CONTINUE
 3690  CONTINUE
       GO TO 4000
C
C ---  ICMPLT=1
C
 3700  SPW=SHT/FIAWT
C
       DO 3705 IMIS=1,NTAR
       DO 3705 J=1,MNWT
 3705  IAWSV(IMIS,J)=IAW(IMIS,J)
C
       DO 3720 IMIS=1,NTAR
       A=RAN(IRS)
       IF(A-PK)3710,3710,3720
 3710  DO 3715 J=1,MNWT
       IAT(IMIS,J)=0
       IAW(IMIS,J)=0
       IAD(IMIS,J,1)=0
 3715  IAD(IMIS,J,2)=0
```

```
 3720   CONTINUE
C
        DO 3803 I=1,NTAR
 3803   INDTA(I)=0
C
        NSHTR=NSHT-NTAR
        ITEMP=0
 3804   IF(ITEMP-NSHTR)3805,3805,3900
 3805   A=RAN(IRS)
        AMIS=A*TAR
        IMIS=JNINT(AMIS)
        IMIS=MAX(IMIS,1)
        IF(INDTA(IMIS)-0)3806,3806,3805
 3806   IAWMT=0
        DO 3810 J=1,MNWT
 3810   IAWMT=IAWMT+IAWSV(IMIS,J)
        FIAWMT=IAWMT
        SHTS=SPW*FIAWMT
        IF(SHTS-1.)3804,3820,3820
 3820   ISHTS=SHTS
        FISHTS=ISHTS
        REM=SHTS-FISHTS
        A=RAN(IRS)
        IF(A-REM)3840,3840,3845
 3840   ISHTST=(ISHTS-1)+1
        GO TO 3850
 3845   ISHTST=(ISHTS-1)
        IF(ISHTST-1)3804,3850,3850
 3850   ITEMP=ITEMP+ISHTST
        INDTA(IMIS)=1
        DO 3880 J=1,ISHTST
        A=RAN(IRS)
        IF(A-PK)3860,3860,3880
 3860   DO 3870 K=1,MNWT
        IAT(IMIS,K)=0
        IAW(IMIS,K)=0
        IAD(IMIS,K,1)=0
 3870   IAD(IMIS,K,2)=0
 3880   CONTINUE
        GO TO 3804
C
 3900   CONTINUE
C
 4000   CONTINUE
        RETURN
        END
```

```
      SUBROUTINE MCD(NVT,NSHT,NTAR,MNWT,PK,IND)
      COMMON/VAL/IV(2533)
      COMMON/DEF/ITD(2533)
      COMMON/ATT/IAT(1500,2),IAW(1500,2)
      COMMON/DEC/IAD(1500,2,2)
      COMMON/RNSEED/IRS
      DIMENSION IAWT(1500,2)
C
      SHT=NSHT
      TAR=NTAR
C
      IDECOYS=0
      PMD=.30
      PFA1=.30
      PFA2=.10
C
      DO 600 I=1,NTAR
      DO 600 J=1,MNWT
  600 IAWT(I,J)=IAW(I,J)
C
      IF(IDECOYS-1)800,5800,5800
C
C --- NO DECOYS
C
  800 IF(IND-2)1000,2000,3000
C
C --- NO DECOYS, RANDOM DEFENSE. RVS CAN BE ENGAGED EVEN IF ALREADY DEAD,
C --- SINCE THERE IS NO COORDINATION
C
 1000 DO 1100 I=1,NSHT
 1005 A=RAN(IRS)
      AMIS=A*TAR
      IMIS=JNINT(AMIS)
      IMIS=MAX(IMIS,1)
      DO 1010 J=1,MNWT
      IF(IAW(IMIS,J)-0)1010,1010,1050
 1010 CONTINUE
      IF(I-NSHT)1005,1100,1100
C
 1050 A=RAN(IRS)
      IF(A-PK)1060,1060,1100
 1060 IF(IAWT(IMIS,J)-0)1070,1070,1065
 1065 IAWT(IMIS,J)=IAWT(IMIS,J)-1
      GO TO 1100
 1070 CONTINUE
C
 1100 CONTINUE
C
      GO TO 9000
C
C --- NO DECOYS, EFFICIENT DEFENSE.
C
 2000 ITEMP=0
      DO 2050 I=1,NTAR
      DO 2050 J=1,MNWT
 2050 ITEMP=ITEMP+IAW(I,J)
      TEMP=ITEMP
```

```
C
      IF(NSHT-ITEMP)2100,2100,2600
C
C ---  SHOOTERS LESS THAN TARGETS. KVS CANNOT BE ENGAGED IF ALREADY DEAD,
C ---  SINCE THERE IS COORDINATION.
C
 2100 DO 2200 I=1,NSHT
 2105 A=RAN(IRS)
      AMIS=A*TAR
      IMIS=JNINT(AMIS)
      IMIS=MAX(IMIS,1)
      DO 2155 J=1,MNWT
      IF(IAWT(IMIS,J)-0)2155,2155,2160
 2155 CONTINUE
      IF(I-NSHT)2105,2200,2200
C
 2160 A=RAN(IRS)
      IF(A-PK)2170,2170,2200
 2170 IAWT(IMIS,J)=IAWT(IMIS,J)-1
C
 2200 CONTINUE
      GO TO 9000
C
C ---  SHOOTERS MORE THAN TARGETS.
C ---  IRP1 TO P1 OF TARGETS
C ---  IR TO (1-P1) OF TARGETS
C
 2600 R=SHT/TEMP
      IR=R
      FIR=IR
      P1=R-FIR
      IRP1=IR+1
C
      DO 2900 I=1,NTAR
      DO 2900 J=1,MNWT
 2610 IF(IAWT(I,J)-0)2900,2900,2620
 2620 IAWTT=IAWT(I,J)
C
      DO 2800 K=1,IAWTT
      A=RAN(IRS)
      IF(A-P1)2650,2650,2750
C
 2650 DO 2670 L=1,IRP1
      A=RAN(IRS)
      IF(A-PK)2680,2680,2670
 2670 CONTINUE
      GO TO 2800
 2680 IAWT(I,J)=IAWT(I,J)-1
      GO TO 2800
C
 2750 DO 2770 L=1,IR
      A=RAN(IRS)
      IF(A-PK)2780,2780,2770
 2770 CONTINUE
      GO TO 2800
 2780 IAWT(I,J)=IAWT(I,J)-1
      GO TO 2800
```

```
C
 2800   CONTINUE
 2900   CONTINUE
C
        GO TO 9000
C
C ---   NO DECOYS, PREFERENTIAL DEFENSE.
C
 3000   CALL MCD3(NVT,NSHT,NTAR,MNWT,PK)
        GO TO 9999
C
C ---   DECOYS
C
 5800   IF(IND-2)6000,7000,8000
C
C ---   DECOYS, RANDOM DEFENSE. RVS CAN BE ENGAGED EVEN IF ALREADY DEAD,
C ---   SINCE THERE IS NO COORDINATION.
C
 6000   DO 6200 I=1,NSHT
 6005   A=RAN(IRS)
        AMIS=A*TAR
        IMIS=JNINT(AMIS)
        IMIS=MAX(IMIS,1)
        DO 6010 J=1,MNWT
        IF(IAW(IMIS,J)-0)6010,6010,6015
 6010   CONTINUE
        IF(I-NSHT)6005,6200,6200
C
 6015   CALL MCDWD(IND,IAWT,IMIS,J,MNWT,PMD,PFA1,PFA2,PK,INDSHT)
        IF(INDSHT-1)6005,6200,6200
C
 6200   CONTINUE
        GO TO 9000
C
C ---   DECOYS, EFFICIENT DEFENSE.
C
 7000   ITEMP=0
        DO 7010 I=1,NTAR
        DO 7010 J=1,MNWT
 7010   ITEMP=ITEMP+IAW(I,J)
        TEMP=ITEMP
C
        IF(NSHT-ITEMP)7100,7100,7600
C
C ---   SHOOTERS LESS THAN TARGETS. RVS CANNOT BE ENGAGED IF ALREADY DEAD,
C ---   SINCE THERE IS COORDINATION.
C
 7100   DO 7200 I=1,NSHT
 7105   A=RAN(IRS)
        AMIS=A*TAR
        IMIS=JNINT(AMIS)
        IMIS=MAX(IMIS,1)
        DO 7110 J=1,MNWT
        IF(IAWT(IMIS,J)-0)7110,7110,7115
 7110   CONTINUE
        IF(I-NSHT)7105,7200,7200
C
```

```
 7105   CALL MCDWD(IND,IAWT,IMIS,J,MNWT,PMD,PFA1,PFA2,PK,INDSHT)
        IF(INDSHT-1)7105,7200,7200
C
 7200   CONTINUE
        GO TO 9000
C
C ---   SHOOTERS MORE THAN TARGETS
C ---   IRP1 TO P1 OF TARGETS
C ---   IR TO (1-P1) OF TARGETS
C
 7500   R=SHT/TEMP
        IR=R
        FIR=IR
        P1=R-FIR
        IRP1=IR+1
C
        DO 7900 I=1,NTAR
        DO 7900 J=1,MNWT
        IF(IAWT(I,J)-0)7900,7900,7620
 7620   IAWTT=IAWT(I,J)
C
        DO 7800 K=1,IAWTT
        A=RAN(IRS)
        IF(A-P1)7650,7650,7750
C
 7650   DO 7670 L=1,IRP1
 7655   CALL MCDWD(IND,IAWT,I,J,MNWT,PMD,PFA1,PFA2,PK,INDSHT)
        IF(INDSHT-1)7655,7670,7670
 7670   CONTINUE
        GO TO 7800
C
 7750   DO 7770 L=1,IR
 7755   CALL MCDWD(IND,IAWT,I,J,MNWT,PMD,PFA1,PFA2,PK,INDSHT)
        IF(INDSHT-1)7755,7770,7770
 7770   CONTINUE
C
 7800   CONTINUE
 7900   CONTINUE
        GO TO 9000
C
C ---   DECOY,PREFERENTIAL DEFENSE.
C
 8000   CONTINUE
C
C ---   SET IAW ARRAY EQUAL IAWT ARRAY
C
 9000   DO 9010 I=1,NTAR
        DO 9010 J=1,MNWT
 9010   IAW(I,J)=IAWT(I,J)
C
 9999   CONTINUE
        RETURN
        END
```

```
      SUBROUTINE MCD3(NVT,NSHT,NTAR,MNWT,PK)
      COMMON/VAL/IV(2533)
      COMMON/DEF/ITD(2533)
      COMMON/ATT/IAT(1500,2),IAW(1500,2)
      COMMON/RNSEED/IRS
      DIMENSION IAWT(1500,2)
      DIMENSION NWT(2533),INDI(2533,3),INDJ(2533,3)
C
 10   FORMAT(4I10,F10.2)
      WRITE(*,10)NVT,NSHT,NTAR,MNWT,PK
C
      LPI=0
      MNWTC=3
C
      DO 500 I=1,NVT
      NWT(I)=0
      DO 500 J=1,MNWTC
      INDI(I,J)=0
 500  INDJ(I,J)=0
      DO 600 I=1,NTAR
      DO 600 J=1,MNWT
 600  IAWT(I,J)=IAW(I,J)
C
C --- IMPACT-POINT PREDICTION
C --- SELECT TARGETS OF MCD TO ASSURE THAT TERMINAL DEFENSE
C --- RECEIVES WHAT IT CAN HANDLE
C
      DO 3260 IMAIN=1,NVT
      JT=0
      DO 3200 I=1,NTAR
      DO 3200 J=1,MNWT
      IF(IAT(I,J)-IMAIN)3200,3100,3200
 3100 NWT(IMAIN)=NWT(IMAIN)+IAWT(I,J)
      JT=JT+1
      INDI(IMAIN,JT)=I
      INDJ(IMAIN,JT)=J
 3200 CONTINUE
C
      IF(LPI-0)3250,3220,3220
C
C --- PRINT INTERMEDIATE CALCULATIONS, NOTE FOR J=2
C
 20   FORMAT(10I10)
 100  FORMAT(1H1,'NWT')
 101  FORMAT(1H1,'INDI(I,1)')
 102  FORMAT(1H1,'INDI(I,2)')
 103  FORMAT(1H1,'INDJ(I,1)')
 104  FORMAT(1H1,'INDJ(I,2)')
 3220 WRITE(6,100)
      WRITE(6,20)(NWT(I),I=1,NVT)
      WRITE(6,101)
      WRITE(6,20)(INDI(I,1),I=1,NVT)
      WRITE(6,102)
      WRITE(6,20)(INDI(I,2),I=1,NVT)
      WRITE(6,103)
      WRITE(6,20)(INDJ(I,1),I=1,NVT)
      WRITE(6,104)
```

```
        WRITE(6,20)(INDJ(I,2),I=1,NVT)
C
 3250   DO 3800 IRATS=1,7
        IF(IRATS-2)3261,3262,3252
 3252   IF(IRATS-4)3263,3264,3253
 3253   IF(IRATS-6)3265,3266,3267
 3261   RATSU=1.
        RATSL=.8
        GO TO 3270
 3262   RATSU=2.
        RATSL=.8
        GO TO 3270
 3263   RATSU=3.
        RATSL=.8
        GO TO 3270
 3264   RATSU=4.
        RATSL=.8
        GO TO 3270
 3265   RATSU=10.
        RATSL=.8
        GO TO 3270
 3266   RATSU=.8
        RATSL=.6
        GO TO 3270
 3267   RATSU=.6
        RATSL=0.
 3270   DO 3500 IMAIN=1,NVT
        IF(NWT(IMAIN)-0)3500,3500,3280
 3280   RAT=NWT(IMAIN)/ITD(IMAIN)
        IF(RAT-RATSU)3300,3300,3500
 3300   DO 3400 J=1,MNWT
        ITEMP=INDI(IMAIN,J)
        JTEMP=INDJ(IMAIN,J)
 3310   IF(IAWT(ITEMP,JTEMP)-0)3400,3400,3320
 3320   NSHT=NSHT-1
        IF(NSHT-0)3600,3325,3325
 3325   A=RAN(IRS)
        IF(A-PK)3330,3330,3320
 3330   IAWT(ITEMP,JTEMP)=IAWT(ITEMP,JTEMP)-1
        NWT(IMAIN)=NWT(IMAIN)-1
        RAT=NWT(IMAIN)/ITD(IMAIN)
        IF(RAT-RATSL)3500,3500,3310
 3400   CONTINUE
 3500   CONTINUE
 3600   CONTINUE
 3800   CONTINUE
C
C ---   SET IAW ARRAY EQUAL TO IAWT ARRAY
C
 4000   DO 4100 I=1,NTAR
        DO 4100 J=1,MNWT
 4100   IAW(I,J)=IAWT(I,J)
C
        CONTINUE
        RETURN
        END
```

```
      SUBROUTINE MCDWD(IND,IAWT,IMIS,JMCD,MNWT,PMD,PFA1,PFA2,PK,INDSHT)
      COMMON/ATT/IAT(1500,2),IAW(1500,2)
      COMMON/DEC/IAD(1500,2,2)
      COMMON/RNSEED/IRS
      DIMENSION IAWT(1500,2)
C
C ---  COUNT OBJECTS ON MISSILE AND CHOOSE OBJECT
C
 6020 NOBJ1=0
      DO 6021 J=1,MNWT
      IF(IND-2)60211,60212,60212
60211 TERM=IAW(IMIS,J)
      GO TO 6021
60212 TERM=IAWT(IMIS,J)
 6021 NOBJ1=NOBJ1+TERM
      NOBJ2=0
      DO 6022 J=1,MNWT
 6022 NOBJ2=NOBJ2+IAD(IMIS,J,1)
      NOBJ3=0
      DO 6023 J=1,MNWT
 6023 NOBJ3=NOBJ3+IAD(IMIS,J,2)
      NOBJT=NOBJ1+NOBJ2+NOBJ3
C
      FNOBJT=NOBJT
      A=RAN(IRS)
      AOBJ=A*FNOBJT
      IOBJ=AOBJ
      IOBJ=MAX(IOBJ,1)
C
C ---  PROCESS OBJECT
C
      IF(IOBJ-NOBJ1)6100,6100,6052
 6052 IF(IOBJ-NOBJ2)6200,6200,6300
C
C ---  OBJECT RV, IF A ELT PMD, DO NOT SHOOT AT IT
C
 6100 A=RAN(IRS)
      IF(A-PMD)6500,6500,6110
 6110 A=RAN(IRS)
      IF(A-PK)6120,6120,6600
 6120 IF(IAWT(IMIS,JMCD)-0)6600,6600,6125
 6125 IAWT(IMIS,JMCD)=IAWT(IMIS,JMCD)-1
      GO TO 6600
C
C ---  OBJECT DECOY, IF A ELT PFA, SHOOT AT IT
C
 6200 PFA=PFA1
      GO TO 6400
 6300 PFA=PFA2
      GO TO 6400
C
 6400 A=RAN(IRS)
      IF(A-PFA)6600,6600,6500
C
C ---  END
C
 6500 INDSHT=0
```

```
0058             GO TO 7000
0059      C
0060       6600  INDSHT=1
0061             GO TO 7000
0062      C
0063       7000  CONTINUE
0064             RETURN
0065             END
```

PROGRAM SECTIONS

```
    Name                                  Bytes   Attributes

  0 $CODE                                   478   PIC CON REL LCL   SHR    EXE   RD NOWRT LONG
  2 $LOCAL                                  108   PIC CON REL LCL NOSHR NOEXE   RD   WRT LONG
  3 ATT                                   24000   PIC OVR REL GBL   SHR NOEXE   RD   WRT LONG
  4 DEC                                   24000   PIC OVR REL GBL   SHR NOEXE   RD   WRT LONG
  5 RNSEED                                    4   PIC OVR REL GBL   SHR NOEXE   RD   WRT LONG

    Total Space Allocated                 48590
```

ENTRY POINTS

```
    Address   Type  Name                References

  0-00000000        MCDWD                  1#
```

VARIABLES

```
    Address   Type  Name        Attributes  References

  2-00000028  R*4   A                         25=     26      37=     38      39=     40      52=     53
  2-0000002C  R*4   AOBJ                      26=     27
  2-00000024  R*4   FNOBJT                    24=     26
 AP-0000000C@ I*4   IHIS                       1      12      14      18      21      41      42(2)
 AP-00000004@ I*4   IND                        1      11

 AP-00000028@ I*4   INDSHT                     1      57=     60=
  2-00000030  I*4   IOBJ                      27=     28(2)=  32      33
  5-00000000  I*4   IRS         COMM           4      25A     37A     39A     52A
  2-00000004  I*4   J                         10=     12      14      17=     18      20=     21
 AP-00000010@ I*4   JMCD                       1      41      42(2)

 AP-00000014@ I*4   MNWT                       1      10      17      20
  2-00000000  I*4   NOBJ1                      9=     15(2)=  22      32
  2-00000010  I*4   NOBJ2                     16=     18(2)=  22      33
  2-00000018  I*4   NOBJ3                     19=     21(2)=  22
  2-00000020  I*4   NOBJT                     22=     24

  2-00000034  R*4   PFA                       47=     49=     53
 AP-0000001C@ R*4   PFA1                       1      47
 AP-00000020@ R*4   PFA2                       1      49
```

```
      SUBROUTINE TD(NVT,NMIS,MNWT,PK,IND)
      COMMON/VAL/IV(2533)
      COMMON/DEF/ITD(2533)
      COMMON/ATT/IAT(1500,2),IAW(1500,2)
      COMMON/RNSEED/IRS
      COMMON/TERM/NW(2533),NWRF(201),NWRMP
      COMMON/INDPRINT/LPI
      DIMENSION JPR(100),KPR(100,50)
C
C ---  IND=1,PREALLOCATED FIXED-SALVO
C ---  IND=2,EFFICIENT
C ---  IND=3,SHOOT-LOOK-SHOOT LIMITED
C ---  IND=4,SHOOT-LOOK-SHOOT UNLIMITED
C

      DO 10 I=1,NVT
 10   NW(I)=0
      DO 20 I=1,201
 20   NWRF(I)=0
C
C ---  NW=NUMBER OF WARHEADS ARRIVING AT EACH TARGET
C
      DO 600 IMAIN=1,NVT
      DO 600 I=1,NMIS
      DO 600 J=1,MNWT
      IF(IAT(I,J)-IMAIN)600,550,600
 550  NW(IMAIN)=NW(IMAIN)+IAW(I,J)
 600  CONTINUE
C
      IF(LPI-1)700,610,700
 605  FORMAT(1H1,'NW BEFORE TD')
 610  WRITE(6,605)
 615  FORMAT(10I10)
      WRITE(6,615)(NW(I),I=1,NVT)
C
C ---  SELECT ATTRITION METHOD
C
 700  IF(IND-2)1000,2000,710
 710  IF(IND-4)3000,4000,4000
C
C ---  IND=1
C ---  PREALLOCATED FIXED-SALVO
C
 1000 CONTINUE
      JPR(2)=2
      JPR(5)=4
      JPR(10)=7
      KPR(2,1)=1
      KPR(2,2)=1
      KPR(5,1)=2
      KPR(5,2)=1
      KPR(5,3)=1
      KPR(5,4)=1
      KPR(10,1)=2
      KPR(10,2)=2
      KPR(10,3)=2
      KPR(10,4)=1
```

```
            KPR(10,5)=1
            KPR(10,6)=1
            KPR(10,7)=1
C
            DO 3800 IMAIN=1,NVT
            IF(NW(IMAIN)-0)3800,3800,3100
      3100  NWT=NW(IMAIN)
            ITDT=ITD(IMAIN)
            JENGS=JPR(ITDT)
            DO 3400 I=1,NWT
            IF(I-JENGS)3150,3150,3800
      3150  KSHTS=KPR(ITDT,I)
            DO 3200 K=1,KSHTS
            A=RAN(IRS)
            IF(A-PK)3170,3170,3200
      3170  NW(IMAIN)=NW(IMAIN)-1
            IF(NW(IMAIN)-0)3800,3800,3400
      3200  CONTINUE
      3400  CONTINUE
      3800  CONTINUE
            GO TO 5000
C
C ---       IND=2
C ---       EFFICIENT
C
      2000  CONTINUE
            DO 4900 IMAIN=1,NVT
            IF(NW(IMAIN)-0)4900,4900,4100
      4100  NWT=NW(IMAIN)
            ITDT=ITD(IMAIN)
            IF(ITDT-NWT)4200,4200,4300
C
C ---       SHOOTERS LESS THAN TARGETS
C
      4200  DO 4250 I=1,ITDT
            A=RAN(IRS)
            IF(A-PK)4210,4210,4250
      4210  NW(IMAIN)=NW(IMAIN)-1
      4250  CONTINUE
            GO TO 4900
C
C ---       SHOOTERS GREATER THAN TARGETS
C
      4300  SHT=ITDT
            TAR=NWT
            R=SHT/TAR
            IR=R
            FIR=IR
            P1=R-FIR
            IRP1=IR+1
C
C ---       IRP1 TO P1 OF TARGETS
C ---       IR TO (1-P1) OF TARGETS
C
            TP1=TAR*P1
            NTP1=JNINT(TP1)
            NTP=NWT-NTP1
```

```
            IF(NTP1-1)4410,4310,4310
      C
      C ---   TARGETS RECEIVING IRP1
      C
       4310   DO 4350 I=1,NTP1
              DO 4340 J=1,IRP1
              A=RAN(IRS)
              IF(A-PK)4320,4320,4340
       4320   NW(IMAIN)=NW(IMAIN)-1
              GO TO 4350
       4340   CONTINUE
       4350   CONTINUE
      C
      C ---   TARGETS RECEIVING IR
      C
       4410   DO 4450 I=1,NTP
              DO 4440 J=1,IR
              A=RAN(IRS)
              IF(A-PK)4420,4420,4440
       4420   NW(IMAIN)=NW(IMAIN)-1
              GO TO 4450
       4440   CONTINUE
       4450   CONTINUE
      C
      C ---   END EFFICIENT
      C
       4900   CONTINUE
              GO TO 5000
      C
      C ---   IND=3
      C ---   SHOOT UP TO MIN(VALUE/5,2) TERMINAL INTERCEPTORS AT EACH INCOMING RV
      C
       3000   CONTINUE
              DO 2800 IMAIN=1,NVT
              IF(NW(IMAIN)-0)2800,2800,2100
       2100   ITDT=ITD(IMAIN)
              NWT=NW(IMAIN)
              IVTEMP=IV(IMAIN)/5
              MAXDEF=MIN0(IVTEMP,2)
              DO 2400 I=1,NWT
              DO 2300 J=1,MAXDEF
              ITDT=ITDT-1
              IF(ITDT-0)2800,2240,2240
       2240   A=RAN(IRS)
              IF(A-PK)2250,2250,2300
       2250   NW(IMAIN)=NW(IMAIN)-1
              IF(NW(IMAIN)-0)2800,2800,2400
       2300   CONTINUE
       2400   CONTINUE
       2800   CONTINUE
              GO TO 5000
      C
      C ---   IND=4
      C ---   SHOOT ITDT TERMINAL INTERCEPTORS ONE BY ONE AT NW(IMAIN) RVS,
      C ---   ELIMINATING RVS AS KILLED.  DEFENSE CAN SHOOT AT EACH RV
      C ---   UNTIL DEFENSE KILLS IT, BUT DEFENSE MAY RUN OUT OF INTERCEPTORS.
      C
```

```
 4000   CONTINUE
        DO 1500 IMAIN=1,NVT
        IF(NW(IMAIN)-0)1500,1500,1305
 1305   ITDT=ITD(IMAIN)
        DO 1400 I=1,ITDT
        A=RAN(IRS)
        IF(A-PK)1310,1310,1400
 1310   NW(IMAIN)=NW(IMAIN)-1
        IF(NW(IMAIN)-0)1500,1500,1400
 1400   CONTINUE
 1500   CONTINUE
        GO TO 5000
C
C ---   CREATE RELATIVE FREQUENCY ARRAY
C
 5000   DO 5200 IMAIN=1,NVT
        IT=NW(IMAIN)
        ITP1=IT+1
 5200   NWRF(ITP1)=NWRF(ITP1)+1
C
        RETURN
        END
```

PROGRAM SECTIONS

Name	Bytes	Attributes
0 $CODE	1552	PIC CON REL LCL SHR EXE RD NOWRT LONG
1 $PDATA	23	PIC CON REL LCL SHR NOEXE RD NOWRT LONG
2 $LOCAL	20580	PIC CON REL LCL NOSHR NOEXE RD WRT LONG
3 VAL	10132	PIC OVR REL GBL SHR NOEXE RD WRT LONG
4 DFF	10132	PIC OVR REL GBL SHR NOEXE RD WRT LONG
5 ATT	24000	PIC OVR REL GBL SHR NOEXE RD WRT LONG
6 RNSEED	4	PIC OVR REL GBL SHR NOEXE RD WRT LONG
7 TERM	10940	PIC OVR REL GBL SHR NOEXE RD WRT LONG
8 INDPRINT	4	PIC OVR REL GBL SHR NOEXE RD WRT LONG
Total Space Allocated	77367	

ENTRY POINTS

Address	Type	Name	References
0-00000000		TD	1#

VARIABLES

Address	Type	Name	Attributes	References							
2-00004FF0	R*4	A		71=	72	93=	94	121=	122	132=	133
				158=	159	177=	178				
2-0000500C	R*4	FIR		105=	106						

```
0001            SUBROUTINE VALSURV(NVT,IVSS)
0002            COMMON/VAL/IV(2533)
0003            COMMON/TERM/NW(2533),NWRF(201),NWRMP
0004      C
0005            IVSS=0
0006            DO 1100 I=1,NVT
0007            IF(NW(I)-0)1050,1050,1100
0008       1050 IVSS=IVSS+IV(I)
0009       1100 CONTINUE
0010      C
0011            CONTINUE
0012            RETURN
0013            END
```

PROGRAM SECTIONS

```
    Name                                 Bytes   Attributes

  0 $CODE                                   71   PIC CON REL LCL   SHR   EXE   RD NOWRT LONG
  2 $LOCAL                                   8   PIC CON REL LCL NOSHR NOEXE   RD   WRT LONG
  3 VAL                                  10132   PIC OVR REL GBL   SHR NOEXE   RD   WRT LONG
  4 TERM                                 10940   PIC OVR REL GBL   SHR NOEXE   RD   WRT LONG

    Total Space Allocated                21151
```

ENTRY POINTS

```
    Address   Type  Name                   References

  0-00000000        VALSURV                  1#
```

VARIABLES

```
    Address   Type  Name          Attributes  References

  2-00000000  I*4   I                            6=       7       8
 AP-00000008@ I*4   IVSS                         1        5=      8(2)=
 AP-00000004@ I*4   NVT                          1        6
  4-00002A68  I*4   NWRMP         COMM           3
```

ARRAYS

```
    Address   Type  Name          Attributes      Bytes  Dimensions       References

  3-00000000  I*4   IV            COMM            10132  (2533)              2       8
  4-00000000  I*4   NW            COMM            10132  (2533)              3       7
  4-00002794  I*4   NWRF          COMM              804  (201)               3
```

```
IVS
    26900
IMAIN
        1
IIDS
     5950
IAWS
    10000
NSHT OF BPD, IND
     1000          2
IAWS AFTER BPD
     3960
NSHT OF MCD, IND
     3000          2
IAWS AFTER MCD
     1269
IND OF TD
                   1
NWS AFTER TD
      435
NWRF
2375    45    38    32    17    13     9     2     1     1     0     0     0     0     0     0     0     0     0     0     0
IVSS,IVSD
    23840       3060
AVES,AVED,SD
   23840.      3060.          0.
```

```
IVS
    26900
IMAIN
        1
ITDS
     5950
IAWS
    10000
NSHT OF BPD, IND
     1000          2
IAWS AFTER BPD
     3960
NSHT OF MCD, IND
     3000          2
IAWS AFTER MCD
     1260
IND OF TD
                   4
NWS AFTER TD
      313
NWRF
2410   37   33   28    8   11    4    1    1    0    0    0    0    0    0    0    0    0    0    0    0
IVSS,IVSD
   24540       2360
AVES,AVED,SD
   24540.      2360.          0.
```

DISTRIBUTION

IDA PAPER P-1966

MONTE CARLO LAYERED DEFENSE MODEL

30 Copies

	Copies
Director, Office of the Joint Chiefs of Staff Washington, D.C. 20301-5000 ATTN: Director, Joint Analysis Directorate	1
Defense Technical Information Center Cameron Station Alexandria, Virginia	2
Deputy Chief of Staff for Research and Development and Acquisition Department of the Army Room 3A474, The Pentagon Washington, D.C. 20310 ATTN: Director, Missile and Air Systems Division RDA/DAMA-WS	1
Office of the Chief of Staff Department of the Army Ballistic Missile Defense Program Office P.O. Box 15280 Arlington, Virginia 22215 ATTN: DACS-BMZ	1
Commander Department of the Army Ballistic Missile Defense Systems Command P.O. Box 1500 Huntsville, Alabama 35807 ATTN: Library	1
Headquarters Department of the Air Force Assistant Chief of Staff Studies and Analysis Room 1E388, The Pentagon Washington, D.C. 20330 ATTN: Library	1
Directorate of Aerospace Studies Deputy Chief of Staff, Plans and Programs Headquarters, Air Force Systems Command Kirtland AFB, NM 89117 ATTN: Library	1

The Rand Corporation
P.O. Box 2138
Santa Monica, California 90406-2138
ATTN: Library 1

The Rand Corporation
2100 M Street, N.W.,
Washington, D.C. 20037
ATTN: Library 1

Hudson Institute, Inc.
Center for Naval Analysis (CNA)
P.O. Box 11280
Alexandria, Virginia 22311
ATTN: Library 1

Los Alamos National Laboratory
P.O. Box 1663, Mail Station 5000
Los Alamos, NW 87545
ATTN: Library 1

University of California
Lawrence Livermore National Laboratory
P.O. Box 808
Livermore, CA. 94550
ATTN: Library 1

Analytic Services, Inc. (ANSER)
Crystal Gateway 3
1215 Jefferson Davis Highway
Arlington, Virginia 22202
ATTN: Library 1

Teledyne-Brown Engineering
Cummings Research Park
Huntsville, Alabama 35807
ATTN: Library 1

Teledyne-Brown Engineering
1250 Academy Park Loop
Colorado Springs, Colorado 80910
ATTN: Library 1

SAIC
1710 Goodridge Drive
McLean, Virginia 22102
ATTN: Library 1

McDonnell-Douglas Astronautics Company
5301 Bolsa Avenue
Huntington Beach, CA. 92647
ATTN: Library 1

Sparta, Inc.
4901 Corporate Drive, Suite 102
Huntsville, Alabama 35805
ATTN: Library 1

System Planning Corporation
1500 Wilson Boulevard
Arlington, Virginia 22209
ATTN: Library 1

Institute for Defense Analyses
1801 North Beauregard Street
Alexandria, Virginia 22311
ATTN: 10

Dr. J. Bracken	2
Dr. W.J. Schultis	1
Mr. R.B. Pirie	1
Mr. S.J. Deitchman	1
Control & Distribution	5